
littlegrid Documentation

Release 2.15.2

littlegrid

Apr 02, 2017

1	Contents	3
1.1	List of examples	3
1.2	Documentation updates	3
1.3	Introduction	4
1.4	Features	5
1.5	Who is using littlegrid	5
1.6	Credits	6
1.7	Getting littlegrid	6
1.8	Getting started with littlegrid	7
1.9	Using Extend proxy	10
1.10	Making tests faster	20
1.11	Fast-start join timeout	21
1.12	No coding needed (launch cluster for C++, .Net and easy playing)	22
1.13	.Net/C++ Extend client tutorial	24
1.14	Failover testing	25
1.15	Multiple autonomous clusters	38
1.16	Maven best practice	39
1.17	Continuous Integration (CI)	41
1.18	Controlling and configuring littlegrid	42
1.19	Controlling system property names used by littlegrid	44
1.20	Reusable cluster member group	46
1.21	Parallel tests	47
1.22	Sample exception report	47
1.23	littlegrid JavaDoc	54
1.24	littlegrid test coverage report	54
1.25	FAQ	54
1.26	Support	55
1.27	License	56
1.28	Release notes	56
1.29	Management REPL application	60



littlegrid

littlegrid is a **small** and **simple** open source framework for Oracle Coherence that lets you run multiple Coherence cluster members in a **single JVM**.

Testing in-process (using single JVM) is a **highly effective** way of developing your Coherence related code because the cluster is **quick to start** and you can **debug locally**.

With just a few lines of littlegrid code you can quickly try out ideas in your **IDE** or easily automate your Coherence tests with your favourite test framework (e.g. **JUnit**) for running on your **Continuous Integration** (CI) build servers.

Coherence + littlegrid = develop faster :-)

Quick code snippet of two cache servers (storage-enabled servers) and then joining the cluster as a storage-disabled client.

```
ClusterMemberGroup memberGroup = ClusterMemberGroupUtils.newBuilder()
    .setStorageEnabledCount(2)
    .buildAndConfigureForStorageDisabledClient(); // this bit is littlegrid

NamedCache cache = CacheFactory.getCache("test");
```


List of examples

The following examples using littlegrid are available (more to be added soon):

- *Getting started with littlegrid* This test will give you a quick impression of littlegrid
- *Using Extend proxy*
- *No coding needed (launch cluster for C++, .Net and easy playing)*
- *Failover testing* This test shows an entry processor failing over and the difference between stop and shutdown.
- *Multiple autonomous clusters* This test shows how to run two separate clusters in a single JVM, this is useful to push-rep testing.

Documentation updates

The below outlines the most recent major changes to the documentation and website, thus allowing visitors to easily determine what changes have occurred since their last visit.

Note: Contact 'help at littlegrid dot org' to suggest website changes or corrections. If you want to contribute code or documentation updates, then let us know - all help is welcome :-)

2015-09-09 Added reference to Coherence cookbook which is using littlegrid for Coherence-related examples and testing, see *Who is using littlegrid*.

2014-12-03 Initial raw example of POM for parallel test execution. Added comment on JavaDoc and test coverage not presently being available due to hosting changes.

2014-11-07 Added new sector to littlegrid users.

2014-09-12 Moving all documentation to readthedocs.org

2014-09-10 Revising and tidying documentation.

2014-07-17 Added a new sector of littlegrid users, see *Who is using littlegrid*.

2013-12-15 Updated configuring and controlling littlegrid, see *Controlling and configuring littlegrid*.

2013-12-15 Start work on replacing-default-cache-server.

2013-12-15 Updated page about making tests faster *Making tests faster* and *Fast-start join timeout*.

2013-11-25 Started work on mini-cluster for Extend *.Net/C++ Extend client tutorial*

2013-11-23 Website went live on faster hosted environment using littlegrid.net domain.

2013-11 Updated content on *Controlling system property names used by littlegrid*.

Introduction

littlegrid is a **small, simple and pragmatic open source framework** that lets you run an entire Oracle Coherence cluster in a **single JVM** for performing developer focused integration tests (often referred to as functional tests).

Note: The littlegrid website is regularly visited by many people from over 70 different countries across the world - lots of people return to find out more, are you interested to know more?

If you answer ‘YES’ to any of the below questions, then littlegrid can help you out:

- Do you want to quickly, easily and effectively test the functionality of your Coherence code in a small cluster? So, do you want to easily test your entry processor/cache store/backing map listeners/interceptors etc. *YES - show me now!*

Warning: Did you know if you run tests with a single cluster member then you aren’t testing properly!

- Do you want to easily and effectively test with an Extend client, Extend proxy server(s) and cache server(s) in literally seconds and have them shutdown all at once when the test stops? *YES - sounds good!*
- Do you want to easily automate your integration or functional tests with your favourite test framework (e.g. JUnit) and run all the tests on a Continuous Integration (CI) server? *YES - very cool!*
- Do you want to test what happens if a cluster member fails over or leaves the cluster? *YES - that could be useful!*

With littlegrid and just a few lines of code, you can quickly and easily try out fully-clustered ideas in your favourite IDE or automate your Coherence integration tests and have them running on your CI (Continuous Integration) server.

littlegrid was developed to facilitate Test Driven Development (TDD) with Coherence and enable integration/functional tests to be easily and reliably run in Continuous Integration (CI) environments.

littlegrid has steadily evolved over a period of time, based upon continuous use on complex projects, along with valuable feedback from in-house Coherence SME (Subject Matter Experts) and developers who are actively using littlegrid on their projects.

littlegrid runs and is used actively used across most of the common platforms: Windows, Mac OSX, Linux and Solaris.

Note: Unit, functional/integration tests with littlegrid have been run on CI servers and locally many, many thousands of times without causing any hangs or problems.

Features

Here's a brief overview of some of littlegrid's features.

- Start, stop and shutdown cluster members.
- Runs in a single JVM.
- Easy way to automate Coherence integration tests.
- Simple, compact and fluent API.
- Test framework agnostic (JUnit/TestNG) – can even be used in standalone program.
- No third party dependencies - just Coherence is required (3.5 - 12.1.3 are supported).
- Run using an IDE, Maven, Ant, Gradle etc. – Continuous Integration (CI) friendly.
- Open source (BSD).
- Fast-start option - helps start the cluster even faster.
- Re-use an existing cluster - helps tests run faster by reducing the number of cluster start-ups and shutdowns.
- Run tests in parallel using multiple forked clusters using a parallel test runner like Surefire.
- Override configuration using system properties or environment variables - useful if you need to precisely control which ports are used for different build plans on your CI (Continuous Integration) server.
- Access the child-first class loader containing the Coherence cluster member - this enables you to control or load classes and objects directly within any of the class loaders running Coherence members.
- Control the class-path used by the Coherence cluster members or easily exclude JARs from the default class-path.
- Exception reporter - produces a report-style output in the event of the cluster not starting, this helps with troubleshooting the problem.
- Easily replace littlegrid's default cluster member if you need to provide hooks for your own life-cycle methods (before or after the cluster member starts and before or after it shuts down).
- Easily substitute your own system property names if you aren't using typical Coherence system property names.

Who is using littlegrid

No one likes to be on their own - so, don't worry you're not! Lots of people are using littlegrid for **trying out ideas**, doing **POCs** or using it on **very complex full-blown projects** with success.

The littlegrid website is regularly visited by many people from over **70 different countries across the world** - lots of people return to find out more, are you interested to know more?

Like any open source project, it is difficult to know precise numbers, but littlegrid is certainly being used in the following industry sectors by companies wanting to easily test their Coherence code:

- Global investment banking
- Global brokerage
- Trade clearing
- Gaming/Gambling
- Deal trading

- Global software vendor
- Global telecommunications
- Open source project
- Video game publishing
- Global retail banking
- Global online travel
- Software house

littlegrid was the chosen Coherence test-support framework for a community Coherence cookbook:

- <http://www.shadowmist.co.uk/index.php/coherence-cookbook>

If using littlegrid and are happy, then get your industry sector added to the list. You don't need to provide the company name, just the industry sector is useful.

If you've tried littlegrid and maybe hit a problem, then contact `help` at `littlegrid dot org` and let's get you heading in the right direction :-)

Credits

littlegrid is developed and maintained by Jon Hall (`jon` at `littlegrid dot org`). littlegrid incorporates many ideas and suggestions from other Coherence Consultants and is actively used by Coherence Consultants because it is easy to use and provides the features required on typical Coherence projects.

[Jon Hall on LinkedIn](#)

Please feel free to drop me an email and let me know what you think about littlegrid :-)



Credit to Jonathan Knight (JK) and Andrew Wilson for original idea of using a separate class-loader for each Coherence cluster member for the purposes of in-process testing.

Website and documentation developed using Sphinx

Website hosting by readthedocs.org

Getting littlegrid

You can get littlegrid in a number of ways - if you use Maven or Gradle, then it is available from Maven central, alternatively you can download the littlegrid JAR from the below link.

- [Download latest stable release](#)
- [Using Maven and *Maven best practice*](#)
- [Browse the latest development source](#)

Note: littlegrid doesn't have any additional third-party dependencies - you just need Coherence and littlegrid to get up and running.

Getting started with littlegrid

This getting started will only take about 1-2 minutes to complete, but by the end you would have run an entire Coherence cluster in a single JVM and performed operations against a cache.

Before getting started, you'll need to download [Coherence](#) and follow *Getting littlegrid* (littlegrid is Maven and Gradle friendly). Once you've got Coherence and littlegrid, you're ready to create a simple JUnit test - you can use your favourite IDE or even Notepad if you want :-).

Note: Here's a JUnit example - note the use of `@BeforeClass` and `@AfterClass`, this enables the cluster to be re-used across the tests in the class.

```
import com.tangosol.net.CacheFactory;
import com.tangosol.net.NamedCache;
import org.junit.AfterClass;
import org.junit.BeforeClass;
import org.junit.Test;
import org.littlegrid.ClusterMemberGroup;
import org.littlegrid.ClusterMemberGroupUtils;

import static org.junit.Assert.assertEquals;

/**
 * Simple integration test example.
 */
public final class SimpleStorageEnabledIntegrationTest {
    private static ClusterMemberGroup memberGroup;

    @BeforeClass
    public static void beforeTests() {
        memberGroup = ClusterMemberGroupUtils.newBuilder()
            .setStorageEnabledCount(2)
            .buildAndConfigureForStorageDisabledClient();
    }

    @AfterClass
    public static void afterTests() {
        ClusterMemberGroupUtils.
↳ shutdownCacheFactoryThenClusterMemberGroups(memberGroup);
    }

    @Test
    public void simpleExample() {
        final NamedCache cache = CacheFactory.getCache("test");
        cache.put("key", "Hello World");

        assertEquals(1, cache.size());
    }
}
```

The above simple test actually does quite a bit with just a couple of lines of code, firstly it starts two cache servers (storage-enabled cluster members) and then the test itself joins the cluster as a storage-disabled client where it then performs simple actions against the test cache.

Note: Notice how the littlegrid specific code is both fluent (i.e. simple to understand) and also there isn't much of it - littlegrid is fairly non-intrusive to introduce into an existing code-base.

You should see some output similar to the below:

```
Sep 10, 2014 6:53:50 PM org.littlegrid.impl.DefaultClusterMemberGroupBuilder <init>
INFO: ___ littlegrid 2.16-SNAPSHOT (http://www.littlegrid.org) - initialised.
↳Builder values: {default file=50, override file=3, 'littlegrid_builder_'
↳environment variables=0, 'LITTLEGRID_BUILDER_' environment variables=0, 'littlegrid.
↳builder.' system properties=2}. Builder to Coherence system property mapping
↳values: {default file=41, override file=0} ___
Sep 10, 2014 6:53:50 PM org.littlegrid.impl.DefaultClusterMemberGroupBuilder
↳buildClusterMembers
INFO: ___ littlegrid 2.16-SNAPSHOT starting - Storage-enabled: 2, Extend proxy: 0,
↳Storage-enabled proxy: 0, JMX: 0, Custom configured: 0 ___
Sep 10, 2014 6:53:50 PM org.littlegrid.impl.DefaultConfigurer
↳getSystemPropertiesForTcmpClusterMember
WARNING: Fast-start join timeout specified. Note: the fast-start Coherence override
↳file will now be configured to be used
Sep 10, 2014 6:53:50 PM org.littlegrid.impl.DefaultClusterMemberGroup
↳outputStartAllMessages
INFO: System properties to be set.: {com.sun.management.jmxremote=false, littlegrid.
↳join.timeout.milliseconds=100, tangosol.coherence.cacheconfig=coherence/littlegrid-
↳test-cache-config.xml, tangosol.coherence.cluster=littlegridCluster, tangosol.
↳coherence.distributed.localstorage=true, tangosol.coherence.extend.address=127.0.0.
↳1, tangosol.coherence.localhost=127.0.0.1, tangosol.coherence.localport=40100,
↳tangosol.coherence.log=stdout, tangosol.coherence.log.level=3, tangosol.coherence.
↳machine=DefaultMachine, tangosol.coherence.management=none, tangosol.coherence.
↳management.remote=true, tangosol.coherence.override=littlegrid/littlegrid-fast-
↳start-coherence-override.xml, tangosol.coherence.rack=DefaultRack, tangosol.
↳coherence.role=DedicatedStorageEnabledMember, tangosol.coherence.site=DefaultSite,
↳tangosol.coherence.tcmp.enabled=true, tangosol.coherence.ttl=0, tangosol.coherence.
↳wka=127.0.0.1, tangosol.coherence.wka.port=40100, tangosol.pof.enabled=false}
Sep 10, 2014 6:53:50 PM org.littlegrid.impl.DefaultClusterMemberGroup
↳outputMemoryMessage
INFO: Max memory: 736MB, current: 736MB, free memory: 720MB
2014-09-10 18:53:50.644/0.803 Oracle Coherence 12.1.3.0.0 <Info> (thread=pool-1-
↳thread-1, member=n/a): Loaded operational configuration from "jar:file:/home/jhall/.
↳m2/repository/com/oracle/coherence/coherence/12.1.3-0-0/coherence-12.1.3-0-0.jar!/
↳tangosol-coherence.xml"
2014-09-10 18:53:50.647/0.806 Oracle Coherence 12.1.3.0.0 <Info> (thread=pool-1-
↳thread-1, member=n/a): Loaded operational overrides from "file:/home/jhall/
↳littlegrid/littlegrid-coherence-testsupport/target/classes/littlegrid/littlegrid-
↳fast-start-coherence-override.xml"

Oracle Coherence Version 12.1.3.0.0 Build 52031
Grid Edition: Development mode
Copyright (c) 2000, 2014, Oracle and/or its affiliates. All rights reserved.

2014-09-10 18:53:50.970/1.129 Oracle Coherence GE 12.1.3.0.0 <Info> (thread=pool-1-
↳thread-1, member=n/a): Loaded cache configuration from "file:/home/jhall/littlegrid/
↳littlegrid-coherence-testsupport/target/test-classes/coherence/littlegrid-test-
↳cache-config.xml"
```

```

2014-09-10 18:53:51.550/1.709 Oracle Coherence GE 12.1.3.0.0 <Info> (thread=pool-1-
↳thread-1, member=n/a): Created cache factory com.tangosol.net.
↳ExtensibleConfigurableCacheFactory
2014-09-10 18:53:52.208/2.367 Oracle Coherence GE 12.1.3.0.0 <Warning> (thread=pool-1-
↳thread-1, member=n/a): Local address "127.0.0.1" is a loopback address; this
↳cluster node will not connect to nodes located on different machines
2014-09-10 18:53:52.250/2.409 Oracle Coherence GE 12.1.3.0.0 <Warning> (thread=pool-1-
↳thread-1, member=n/a): UnicastUdpSocket failed to set send buffer size to 8 packets
↳(512KB); actual size is 40%, 3 packets (208KB). Consult your OS documentation
↳regarding increasing the maximum socket buffer size. Proceeding with the actual
↳value may cause sub-optimal performance.
2014-09-10 18:53:52.250/2.409 Oracle Coherence GE 12.1.3.0.0 <Warning> (thread=pool-1-
↳thread-1, member=n/a): UnicastUdpSocket failed to set receive buffer size to 64
↳packets (4MB); actual size is 5%, 3 packets (208KB). Consult your OS documentation
↳regarding increasing the maximum socket buffer size. Proceeding with the actual
↳value may cause sub-optimal performance.
2014-09-10 18:53:52.251/2.410 Oracle Coherence GE 12.1.3.0.0 <Warning> (thread=pool-1-
↳thread-1, member=n/a): PreferredUnicastUdpSocket failed to set receive buffer size
↳to 128 packets (8MB); actual size is 2%, 3 packets (208KB). Consult your OS
↳documentation regarding increasing the maximum socket buffer size. Proceeding with
↳the actual value may cause sub-optimal performance.
2014-09-10 18:53:52.527/2.686 Oracle Coherence GE 12.1.3.0.0 <Info> (thread=Cluster,
↳member=n/a): Created a new cluster "littlegridCluster" with Member(Id=1,
↳Timestamp=2014-09-10 18:53:52.302, Address=127.0.0.1:40100, MachineId=30438,
↳Location=site:DefaultSite,rack:DefaultRack,machine:DefaultMachine,process:32042,
↳Role=DedicatedStorageEnabledMember, Edition=Grid Edition, Mode=Development,
↳CpuCount=4, SocketCount=2)
2014-09-10 18:53:52.529/2.688 Oracle Coherence GE 12.1.3.0.0 <Info> (thread=pool-1-
↳thread-1, member=n/a): Started cluster Name=littlegridCluster

WellKnownAddressList (Size=1,
  WKA{Address=127.0.0.1, Port=40100}
)

MasterMemberSet (
  ThisMember=Member (Id=1, Timestamp=2014-09-10 18:53:52.302, Address=127.0.0.1:40100,
↳MachineId=30438, Location=site:DefaultSite,rack:DefaultRack,machine:DefaultMachine,
↳process:32042, Role=DedicatedStorageEnabledMember)
  OldestMember=Member (Id=1, Timestamp=2014-09-10 18:53:52.302, Address=127.0.0.
↳1:40100, MachineId=30438, Location=site:DefaultSite,rack:DefaultRack,
↳machine:DefaultMachine,process:32042, Role=DedicatedStorageEnabledMember)
  ActualMemberSet=MemberSet (Size=1
    Member (Id=1, Timestamp=2014-09-10 18:53:52.302, Address=127.0.0.1:40100,
↳MachineId=30438, Location=site:DefaultSite,rack:DefaultRack,machine:DefaultMachine,
↳process:32042, Role=DedicatedStorageEnabledMember)
  )
  MemberId|ServiceVersion|ServiceJoined|MemberState
  1|12.1.3|2014-09-10 18:53:52.302|JOINED
  RecycleMillis=1200000
  RecycleSet=MemberSet (Size=0
  )
)

TcpRing{Connections=[]}
IpMonitor{Addresses=0, Timeout=15s}

...
Sep 10, 2014 6:53:55 PM org.littlegrid.impl.DefaultClusterMemberGroupBuilder
↳buildClusterMembers

```

```
INFO: ___ Group of 2 cluster member(s) started in 5177ms, member Ids are: [1, 2] ___
Sep 10, 2014 6:53:55 PM org.littlegrid.impl.DefaultConfigurer
↳getSystemPropertiesForTcmpClusterMember
WARNING: Fast-start join timeout specified. Note: the fast-start Coherence override
↳file will now be configured to be used
Sep 10, 2014 6:53:55 PM org.littlegrid.impl.DefaultConfigurer configureFor
INFO: System properties set for client/member: {com.sun.management.jmxremote=false,
↳littlegrid.join.timeout.milliseconds=100, tangosol.coherence.cacheconfig=coherence/
↳littlegrid-test-cache-config.xml, tangosol.coherence.cluster=littlegridCluster,
↳tangosol.coherence.distributed.localstorage=false, tangosol.coherence.extend.
↳address=127.0.0.1, tangosol.coherence.extend.enabled=false, tangosol.coherence.
↳localhost=127.0.0.1, tangosol.coherence.localport=40100, tangosol.coherence.
↳log=stdout, tangosol.coherence.log.level=3, tangosol.coherence.
↳machine=DefaultMachine, tangosol.coherence.management=none, tangosol.coherence.
↳management.remote=true, tangosol.coherence.override=littlegrid/littlegrid-fast-
↳start-coherence-override.xml, tangosol.coherence.rack=DefaultRack, tangosol.
↳coherence.role=StorageDisabledClient, tangosol.coherence.site=DefaultSite, tangosol.
↳coherence.tcmp.enabled=true, tangosol.coherence.ttl=0, tangosol.coherence.wka=127.0.
↳0.1, tangosol.coherence.wka.port=40100, tangosol.pof.enabled=false}
Sep 10, 2014 6:53:57 PM org.littlegrid.ClusterMemberGroupUtils
↳shutdownCacheFactoryThenClusterMemberGroups
INFO: About to shutdown cache factory
Sep 10, 2014 6:53:58 PM org.littlegrid.impl.DefaultClusterMemberGroup shutdownAll
INFO: Restoring system properties back to their original state before member group
↳started
2014-09-10 18:53:58.135/8.294 Oracle Coherence GE 12.1.3.0.0 <Info>
↳(thread=DistributedCache, member=n/a): Remains to transfer before shutting down:
↳129 primary partitions, 128 backup partitions
2014-09-10 18:53:58.264/8.423 Oracle Coherence GE 12.1.3.0.0 <Info>
↳(thread=DistributedCache, member=n/a): Remains to transfer before shutting down: 0
↳primary partitions, 128 backup partitions
Sep 10, 2014 6:54:00 PM org.littlegrid.impl.DefaultClusterMemberGroup shutdownAll
INFO: ___ Group of 2 cluster member(s) shutdown in 2392ms ___
```

Notice how littlegrid has done the following:

- Ensured that WKA (Well Known Addresses) are being used and that the network traffic is restricted to this machine - this prevents accidental clustering.
- Given the members sensible rolenames, such as DedicatedStorageEnabledMember.
- Used different port numbers to avoid clustering with vanilla Coherence instances you might have already running.

Note: littlegrid has ways of making tests start faster and re-using an existing cluster across tests for even faster testing.

Using Extend proxy

If you haven't already had a look, the *Getting started with littlegrid* is the best place to start looking at littlegrid.

1) This example shows you how to run two storage-enabled members, an Extend proxy server and connect as an Extend client - all in a single JVM using littlegrid. Save the below class in a file called: SimpleExtendIntegrationTest.java

```
import com.tangosol.net.CacheFactory;
import com.tangosol.net.NamedCache;
```

```

import com.tangosol.util.aggregator.Count;
import com.tangosol.util.filter.AlwaysFilter;
import org.junit.AfterClass;
import org.junit.Before;
import org.junit.BeforeClass;
import org.junit.Test;
import org.littlegrid.ClusterMemberGroup;
import org.littlegrid.ClusterMemberGroupUtils;

import java.util.Collections;

import static junit.framework.Assert.assertEquals;

/**
 * Simple Extend littlegrid integration tests.
 */
public class SimpleExtendIntegrationTest {
    private static final int NUMBER_OF_ITEMS = 250;
    private static ClusterMemberGroup memberGroup;

    private final NamedCache cache = CacheFactory.getCache("simple-extend-example");

    /**
     * Use BeforeClass to start the cluster up before any of the tests run - this
     ↪ensures
     * we only have the start-up delay only once.
     *
     * Note: apart from starting up and shutting down littlegrid, its code and API
     ↪shouldn't
     * really be in any of your tests, unless you want to perform a stop or shutdown
     ↪of a
     * particular member for failover testing.
     */
    @BeforeClass
    public static void beforeTests() {
        // Build the cluster
        //
        // Note: use of the fast-start option and log level set to 6, so that we see
     ↪the
        // interesting Extend logging of client connections.
        //
        memberGroup = ClusterMemberGroupUtils.newBuilder()
            .setStorageEnabledCount(2)
            .setCacheConfiguration("simple-extend-cache-config.xml")
            .setExtendProxyCount(1)
            .setClientCacheConfiguration("simple-extend-client-cache-config.xml")
            .setFastStartJoinTimeoutMilliseconds(1000)
            .setLogLevel(6)
            .buildAndConfigureForExtendClient();
    }

    /**
     * Shutdown the cluster, this method also does a CacheFactory.shutdown() for the
     ↪client
     * to ensure that leaves nicely as well.
     */
    @AfterClass

```

```

    public static void afterTests() {
        ClusterMemberGroupUtils.
↪ shutdownCacheFactoryThenClusterMemberGroups(memberGroup);
    }

    /**
     * For this example test we are clearing down the cache each time and
↪ repopulating it.
     */
    @Before
    public void beforeTest() {
        cache.clear();

        for (int i = 0; i < NUMBER_OF_ITEMS; i++) {
            cache.putAll(Collections.singletonMap(i, i));
        }

        assertEquals(NUMBER_OF_ITEMS, cache.size());
    }

    @Test
    public void putGet() {
        cache.put("key", "value");

        assertEquals(NUMBER_OF_ITEMS + 1, cache.size());
    }

    @Test
    public void aggregateCount() {
        assertEquals(NUMBER_OF_ITEMS, cache.aggregate(AlwaysFilter.INSTANCE, new
↪ Count()));
    }
}

```

2) Before running the test, you need the cache server (storage-enabled) and Extend proxy configuration file. Save the below cache configuration in a file called: simple-extend-cache-config.xml

```

<cache-config>
  <caching-scheme-mapping>
    <cache-mapping>
      <cache-name>*</cache-name>
      <scheme-name>simple-distributed-scheme</scheme-name>
    </cache-mapping>
  </caching-scheme-mapping>

  <caching-schemes>
    <distributed-scheme>
      <scheme-name>simple-distributed-scheme</scheme-name>
      <backing-map-scheme>
        <local-scheme/>
      </backing-map-scheme>
      <autostart>true</autostart>
    </distributed-scheme>

    <!--
      littlegrid uses system properties to enable Extend and control the
↪ address and
      port number that the Extend proxy server runs on.
    -->

```



```

    Note: if you want to use different system property names, that is fine as
↪ littlegrid
    supports a mapping to your system property names.
    -->
    <proxy-scheme>
      <scheme-name>extend-tcp</scheme-name>
      <service-name>ExtendProxyServerService</service-name>
      <acceptor-config>
        <tcp-acceptor>
          <local-address>
↪ 127.x.2.3</address>
          <port system-property="tangosol.coherence.extend.port">9099</
↪ port>
          </local-address>
        </tcp-acceptor>
      </acceptor-config>
      <autostart system-property="tangosol.coherence.extend.enabled">>false</
↪ autostart>
    </proxy-scheme>
  </caching-schemes>
</cache-config>

```

3) Finally you just need the Extend client cache configuration, save this in a file called: simple-extend-client-cache-config.xml

```

<cache-config>
  <caching-scheme-mapping>
    <cache-mapping>
      <cache-name>*</cache-name>
      <scheme-name>extend-tcp</scheme-name>
    </cache-mapping>
  </caching-scheme-mapping>

  <caching-schemes>
    <remote-cache-scheme>
      <scheme-name>extend-tcp</scheme-name>
      <service-name>ExtendProxyClientService</service-name>
      <initiator-config>
        <tcp-initiator>
          <remote-addresses>
            <socket-address>
↪ address">127.x.2.3</address>
            <port system-property="tangosol.coherence.extend.port">
↪ 9099</port>
            </socket-address>
          </remote-addresses>
        </tcp-initiator>
      </initiator-config>
    </remote-cache-scheme>
  </caching-schemes>
</cache-config>

```

4) Now just run the example, try it from your IDE :-). What you'll notice is that littlegrid has set all the system properties and ports numbers, you don't need to do anything special - just concentrate on the Coherence bit and let littlegrid start and shutdown the cluster.

```

Sep 11, 2014 8:47:41 AM org.littlegrid.impl.DefaultClusterMemberGroupBuilder <init>
INFO: ___ littlegrid 2.16-SNAPSHOT (http://www.littlegrid.net) - initialised.
↳Builder values: {default file=50, override file=3, 'littlegrid_builder_'
↳environment variables=0, 'LITTLEGRID_BUILDER_' environment variables=0, 'littlegrid.
↳builder.' system properties=2}. Builder to Coherence system property mapping.
↳values: {default file=41, override file=0} ___
Sep 11, 2014 8:47:41 AM org.littlegrid.impl.DefaultClusterMemberGroupBuilder
↳buildClusterMembers
INFO: ___ littlegrid 2.16-SNAPSHOT starting - Storage-enabled: 2, Extend proxy: 1,
↳Storage-enabled proxy: 0, JMX: 0, Custom configured: 0 ___
Sep 11, 2014 8:47:41 AM org.littlegrid.impl.DefaultConfigurer
↳getSystemPropertiesForTcmpClusterMember
WARNING: Fast-start join timeout specified. Note: the fast-start Coherence override
↳file will now be configured to be used
Sep 11, 2014 8:47:41 AM org.littlegrid.impl.DefaultClusterMemberGroup
↳outputStartAllMessages
INFO: System properties to be set.: {com.sun.management.jmxremote=false, littlegrid.
↳join.timeout.milliseconds=1000, tangosol.coherence.cacheconfig=simple-extend-cache-
↳config.xml, tangosol.coherence.cluster=littlegridCluster, tangosol.coherence.
↳distributed.localstorage=true, tangosol.coherence.extend.address=127.0.0.1,
↳tangosol.coherence.localhost=127.0.0.1, tangosol.coherence.localport=40100,
↳tangosol.coherence.log=stdout, tangosol.coherence.log.level=6, tangosol.coherence.
↳machine=DefaultMachine, tangosol.coherence.management=none, tangosol.coherence.
↳management.remote=true, tangosol.coherence.override=littlegrid/littlegrid-fast-
↳start-coherence-override.xml, tangosol.coherence.rack=DefaultRack, tangosol.
↳coherence.role=DedicatedStorageEnabledMember, tangosol.coherence.site=DefaultSite,
↳tangosol.coherence.tcmp.enabled=true, tangosol.coherence.ttl=0, tangosol.coherence.
↳wka=127.0.0.1, tangosol.coherence.wka.port=40100, tangosol.pof.enabled=false}
Sep 11, 2014 8:47:41 AM org.littlegrid.impl.DefaultClusterMemberGroup
↳outputMemoryMessage
INFO: Max memory: 736MB, current: 736MB, free memory: 724MB
2014-09-11 08:47:42.197/0.938 Oracle Coherence 12.1.3.0.0 <Info> (thread=pool-1-
↳thread-1, member=n/a): Loaded operational configuration from "jar:file:/home/jhall/.
↳m2/repository/com/oracle/coherence/coherence/12.1.3-0-0/coherence-12.1.3-0-0.jar!/
↳tangosol-coherence.xml"
2014-09-11 08:47:42.200/0.941 Oracle Coherence 12.1.3.0.0 <Info> (thread=pool-1-
↳thread-1, member=n/a): Loaded operational overrides from "file:/home/jhall/
↳littlegrid/littlegrid-coherence-testsupport/target/classes/littlegrid/littlegrid-
↳fast-start-coherence-override.xml"
2014-09-11 08:47:42.210/0.951 Oracle Coherence 12.1.3.0.0 <D5> (thread=pool-1-thread-
↳1, member=n/a): Optional configuration override "cache-factory-config.xml" is not
↳specified
2014-09-11 08:47:42.221/0.962 Oracle Coherence 12.1.3.0.0 <D5> (thread=pool-1-thread-
↳1, member=n/a): Optional configuration override "cache-factory-builder-config.xml"
↳is not specified
2014-09-11 08:47:42.224/0.965 Oracle Coherence 12.1.3.0.0 <D5> (thread=pool-1-thread-
↳1, member=n/a): Optional configuration override "/custom-mbeans.xml" is not
↳specified
2014-09-11 08:47:42.225/0.966 Oracle Coherence 12.1.3.0.0 <D6> (thread=pool-1-thread-
↳1, member=n/a): Loaded edition data from "jar:file:/home/jhall/.m2/repository/com/
↳oracle/coherence/coherence/12.1.3-0-0/coherence-12.1.3-0-0.jar!/coherence-grid.xml"

Oracle Coherence Version 12.1.3.0.0 Build 52031
Grid Edition: Development mode
Copyright (c) 2000, 2014, Oracle and/or its affiliates. All rights reserved.

2014-09-11 08:47:42.373/1.114 Oracle Coherence GE 12.1.3.0.0 <Info> (thread=pool-1-
↳thread-1, member=n/a): Loaded cache configuration from "file:/home/jhall/littlegrid/
↳littlegrid-coherence-testsupport/target/test-classes/simple-extend-cache-config.xml
↳"; this document does not refer to any schema definition and has not been validated.

```

```

2014-09-11 08:47:42.598/1.339 Oracle Coherence GE 12.1.3.0.0 <Info> (thread=pool-1-
↳thread-1, member=n/a): Created cache factory com.tangosol.net.
↳ExtensibleConfigurableCacheFactory
2014-09-11 08:47:42.975/1.716 Oracle Coherence GE 12.1.3.0.0 <Warning> (thread=pool-1-
↳thread-1, member=n/a): Local address "127.0.0.1" is a loopback address; this
↳cluster node will not connect to nodes located on different machines
2014-09-11 08:47:43.018/1.759 Oracle Coherence GE 12.1.3.0.0 <Warning> (thread=pool-1-
↳thread-1, member=n/a): UnicastUdpSocket failed to set send buffer size to 8 packets
↳(512KB); actual size is 40%, 3 packets (208KB). Consult your OS documentation
↳regarding increasing the maximum socket buffer size. Proceeding with the actual
↳value may cause sub-optimal performance.
2014-09-11 08:47:43.018/1.759 Oracle Coherence GE 12.1.3.0.0 <Warning> (thread=pool-1-
↳thread-1, member=n/a): UnicastUdpSocket failed to set receive buffer size to 64
↳packets (4MB); actual size is 5%, 3 packets (208KB). Consult your OS documentation
↳regarding increasing the maximum socket buffer size. Proceeding with the actual
↳value may cause sub-optimal performance.
2014-09-11 08:47:43.019/1.760 Oracle Coherence GE 12.1.3.0.0 <Warning> (thread=pool-1-
↳thread-1, member=n/a): PreferredUnicastUdpSocket failed to set receive buffer size
↳to 128 packets (8MB); actual size is 2%, 3 packets (208KB). Consult your OS
↳documentation regarding increasing the maximum socket buffer size. Proceeding with
↳the actual value may cause sub-optimal performance.
2014-09-11 08:47:43.019/1.760 Oracle Coherence GE 12.1.3.0.0 <D4> (thread=pool-1-
↳thread-1, member=n/a): TCMP bound to /127.0.0.1:40100 using
↳SystemDatagramSocketProvider
2014-09-11 08:47:44.304/3.045 Oracle Coherence GE 12.1.3.0.0 <Info> (thread=Cluster,
↳member=n/a): Created a new cluster "littlegridCluster" with Member(Id=1,
↳Timestamp=2014-09-11 08:47:43.072, Address=127.0.0.1:40100, MachineId=30438,
↳Location=site:DefaultSite,rack:DefaultRack,machine:DefaultMachine,process:1473,
↳Role=DedicatedStorageEnabledMember, Edition=Grid Edition, Mode=Development,
↳CpuCount=4, SocketCount=2)
2014-09-11 08:47:44.306/3.048 Oracle Coherence GE 12.1.3.0.0 <Info> (thread=pool-1-
↳thread-1, member=n/a): Started cluster Name=littlegridCluster

WellKnownAddressList (Size=1,
  WKA{Address=127.0.0.1, Port=40100}
)

MasterMemberSet (
  ThisMember=Member(Id=1, Timestamp=2014-09-11 08:47:43.072, Address=127.0.0.1:40100,
↳MachineId=30438, Location=site:DefaultSite,rack:DefaultRack,machine:DefaultMachine,
↳process:1473, Role=DedicatedStorageEnabledMember)
  OldestMember=Member(Id=1, Timestamp=2014-09-11 08:47:43.072, Address=127.0.0.
↳1:40100, MachineId=30438, Location=site:DefaultSite,rack:DefaultRack,
↳machine:DefaultMachine,process:1473, Role=DedicatedStorageEnabledMember)
  ActualMemberSet=MemberSet (Size=1
    Member(Id=1, Timestamp=2014-09-11 08:47:43.072, Address=127.0.0.1:40100,
↳MachineId=30438, Location=site:DefaultSite,rack:DefaultRack,machine:DefaultMachine,
↳process:1473, Role=DedicatedStorageEnabledMember)
  )
  MemberId|ServiceVersion|ServiceJoined|MemberState
  1|12.1.3|2014-09-11 08:47:43.072|JOINED
  RecycleMillis=1200000
  RecycleSet=MemberSet (Size=0
  )
)

...
2014-09-11 08:47:48.774/7.515 Oracle Coherence GE 12.1.3.0.0 <Info>
↳(thread=Proxy:ExtendProxyServerService:TcpAcceptor:TcpProcessor, member=3): The
↳specified local address "/127.0.0.1" is a loopback address; clients running on
↳remote machines will not be able to connect to this TcpAcceptor

```

```

2014-09-11 08:47:48.774/7.515 Oracle Coherence GE 12.1.3.0.0 <Info>
↳ (thread=Proxy:ExtendProxyServerService:TcpAcceptor, member=3): TcpAcceptor now
↳ listening for connections on 127.0.0.1:40150
2014-09-11 08:47:48.778/7.519 Oracle Coherence GE 12.1.3.0.0 <D6>
↳ (thread=Proxy:ExtendProxyServerService:TcpAcceptor, member=3): Started: TcpAcceptor
↳ {Name=Proxy:ExtendProxyServerService:TcpAcceptor, State=(SERVICE_STARTED),
↳ ThreadCount=8, HungThreshold=0, TaskTimeout=0, Codec=Codec(Format=POF),
↳ Serializer=com.tangosol.io.DefaultSerializer, PingInterval=0, PingTimeout=0,
↳ RequestTimeout=0, MaxIncomingMessageSize=0, MaxOutgoingMessageSize=0,
↳ SocketProvider=com.oracle.common.internal.net.MultiplexedSocketProvider@75ee2516,
↳ LocalAddress=WrapperSocketAddressProvider(Providers=[[/127.0.0.1:40150]]),
↳ SocketOptions{LingerTimeout=0, KeepAliveEnabled=true, TcpDelayEnabled=false},
↳ ListenBacklog=0, BufferPoolIn=BufferPool(BufferSize=2KB, BufferType=DIRECT,
↳ Capacity=Unlimited), BufferPoolOut=BufferPool(BufferSize=2KB, BufferType=DIRECT,
↳ Capacity=Unlimited)}
2014-09-11 08:47:48.787/7.528 Oracle Coherence GE 12.1.3.0.0 <D5>
↳ (thread=Proxy:ExtendProxyServerService, member=3): Service ExtendProxyServerService
↳ joined the cluster with senior service member 3
2014-09-11 08:47:52.413/11.154 Oracle Coherence GE 12.1.3.0.0 <D6>
↳ (thread=ExtendProxyClientService:TcpInitiator, member=n/a): Closed:
↳ Channel(Id=1451514861, Open=false)
2014-09-11 08:47:52.413/11.154 Oracle Coherence GE 12.1.3.0.0 <D6>
↳ (thread=ExtendProxyClientService:TcpInitiator, member=n/a): Closed:
↳ Channel(Id=1877724955, Open=false)
2014-09-11 08:47:52.414/11.155 Oracle Coherence GE 12.1.3.0.0 <D6>
↳ (thread=ExtendProxyClientService:TcpInitiator, member=n/a): Closed:
↳ TcpConnection(Id=0x0000014863AEBE76408233D2B27BE1D22C2AF99E33B2B46780DAC1C45705E036,
↳ Open=false, Member(Id=0, Timestamp=2014-09-11 08:47:49.45, Address=127.0.0.1:0,
↳ MachineId=0, Location=site:,machine:localhost,process:1473, Role=ExtendClient),
↳ LocalAddress=127.0.0.1:35206, RemoteAddress=127.0.0.1:40150)
2014-09-11 08:47:52.414/11.155 Oracle Coherence GE 12.1.3.0.0 <D6>
↳ (thread=ExtendProxyClientService:TcpInitiator, member=n/a): Stopped: TcpInitiator
↳ {Name=ExtendProxyClientService:TcpInitiator, State=(SERVICE_STOPPED), ThreadCount=0,
↳ Codec=Codec(Format=POF), Serializer=com.tangosol.io.DefaultSerializer,
↳ PingInterval=0, PingTimeout=0, RequestTimeout=0, MaxIncomingMessageSize=0,
↳ MaxOutgoingMessageSize=0, ConnectTimeout=0, SocketProvider=com.oracle.common.
↳ internal.net.MultiplexedSocketProvider@3458faae,
↳ RemoteAddresses=WrapperSocketAddressProvider(Providers=[[/127.0.0.1:40150]]),
↳ SocketOptions{LingerTimeout=0, KeepAliveEnabled=true, TcpDelayEnabled=false}}
2014-09-11 08:47:52.418/11.159 Oracle Coherence GE 12.1.3.0.0 <D6>
↳ (thread=Proxy:ExtendProxyServerService:TcpAcceptor, member=3): Closed:
↳ Channel(Id=1451514861, Open=false)
2014-09-11 08:47:52.422/11.163 Oracle Coherence GE 12.1.3.0.0 <D6>
↳ (thread=Proxy:ExtendProxyServerService:TcpAcceptor, member=3): Closed:
↳ Channel(Id=1877724955, Open=false)
2014-09-11 08:47:52.423/11.164 Oracle Coherence GE 12.1.3.0.0 <D6>
↳ (thread=Proxy:ExtendProxyServerService:TcpAcceptor, member=3): Closed:
↳ TcpConnection(Id=0x0000014863AEBE76408233D2B27BE1D22C2AF99E33B2B46780DAC1C45705E036,
↳ Open=false, Member(Id=0, Timestamp=2014-09-11 08:47:49.45, Address=127.0.0.1:0,
↳ MachineId=0, Location=site:,machine:localhost,process:1473, Role=ExtendClient),
↳ LocalAddress=127.0.0.1:40150, RemoteAddress=127.0.0.1:35206)
2014-09-11 08:47:52.427/11.168 Oracle Coherence GE 12.1.3.0.0 <D6>
↳ (thread=Proxy:ExtendProxyServerService:TcpAcceptor:TcpProcessor, member=3):
↳ Released:
↳ TcpConnection(Id=0x0000014863AEBE76408233D2B27BE1D22C2AF99E33B2B46780DAC1C45705E036,
↳ Open=false, Member(Id=0, Timestamp=2014-09-11 08:47:49.45, Address=127.0.0.1:0,
↳ MachineId=0, Location=site:,machine:localhost,process:1473, Role=ExtendClient),
↳ LocalAddress=127.0.0.1:40150, RemoteAddress=127.0.0.1:35206)

```

```

2014-09-11 08:47:52.430/11.171 Oracle Coherence GE 12.1.3.0.0 <Info>
↳(thread=DistributedCache, member=n/a): Remains to transfer before shutting down:
↳129 primary partitions, 128 backup partitions
Sep 11, 2014 8:47:52 AM org.littlegrid.ClusterMemberGroupUtils
↳shutdownCacheFactoryThenClusterMemberGroups
INFO: About to shutdown cache factory
Sep 11, 2014 8:47:52 AM org.littlegrid.impl.DefaultClusterMemberGroup shutdownAll
INFO: Restoring system properties back to their original state before member group
↳started
2014-09-11 08:47:52.432/11.173 Oracle Coherence GE 12.1.3.0.0 <D5>
↳(thread=DistributedCache, member=2): This member has become the distribution
↳coordinator for MemberSet (Size=2
  Member (Id=1, Timestamp=2014-09-11 08:47:43.072, Address=127.0.0.1:40100,
↳MachineId=30438, Location=site:DefaultSite,rack:DefaultRack,machine:DefaultMachine,
↳process:1473, Role=DedicatedStorageEnabledMember)
  Member (Id=2, Timestamp=2014-09-11 08:47:46.106, Address=127.0.0.1:40102,
↳MachineId=30438, Location=site:DefaultSite,rack:DefaultRack,machine:DefaultMachine,
↳process:1473, Role=DedicatedStorageEnabledMember)
)
2014-09-11 08:47:52.448/11.189 Oracle Coherence GE 12.1.3.0.0 <D6>
↳(thread=DistributedCache, member=2): SimpleAssignmentStrategy suggested the
↳ownership of Ownership{Owners=2, 0} for PartitionSet{0..256}
2014-09-11 08:47:52.450/11.191 Oracle Coherence GE 12.1.3.0.0 <D4>
↳(thread=DistributedCache, member=2): Asking member 1 for primary ownership of
↳PartitionSet{128..256}
2014-09-11 08:47:52.476/11.217 Oracle Coherence GE 12.1.3.0.0 <D5>
↳(thread=DistributedCache, member=n/a): Transferring primary PartitionSet{128..256}
↳to member 2 requesting 129
2014-09-11 08:47:52.542/11.283 Oracle Coherence GE 12.1.3.0.0 <Info>
↳(thread=DistributedCache, member=n/a): Remains to transfer before shutting down: 0
↳primary partitions, 128 backup partitions
2014-09-11 08:47:52.543/11.284 Oracle Coherence GE 12.1.3.0.0 <D5>
↳(thread=DistributedCache, member=n/a): Primary[]#000
Backup[1]#128: 000, 001, 002, 003, 004, 005, 006, 007, 008, 009, 010, 011, 012, 013,
↳014, 015, 016, 017, 018, 019, 020, 021, 022, 023, 024, 025, 026, 027, 028, 029, 030,
↳031, 032, 033, 034, 035, 036, 037, 038, 039, 040, 041, 042, 043, 044, 045, 046,
↳047, 048, 049, 050, 051, 052, 053, 054, 055, 056, 057, 058, 059, 060, 061, 062, 063,
↳064, 065, 066, 067, 068, 069, 070, 071, 072, 073, 074, 075, 076, 077, 078, 079,
↳080, 081, 082, 083, 084, 085, 086, 087, 088, 089, 090, 091, 092, 093, 094, 095, 096,
↳097, 098, 099, 100, 101, 102, 103, 104, 105, 106, 107, 108, 109, 110, 111, 112,
↳113, 114, 115, 116, 117, 118, 119, 120, 121, 122, 123, 124, 125, 126, 127
2014-09-11 08:47:54.543/13.284 Oracle Coherence GE 12.1.3.0.0 <D5>
↳(thread=DistributedCache, member=n/a): Service DistributedCache left the cluster
2014-09-11 08:47:54.547/13.288 Oracle Coherence GE 12.1.3.0.0 <D5>
↳(thread=DistributedCache, member=3): Member 1 left service DistributedCache with
↳senior member 1
2014-09-11 08:47:54.548/13.289 Oracle Coherence GE 12.1.3.0.0 <D5>
↳(thread=DistributedCache, member=2): Member 1 left service DistributedCache with
↳senior member 1
2014-09-11 08:47:54.548/13.289 Oracle Coherence GE 12.1.3.0.0 <D6>
↳(thread=DistributedCache, member=2): Service DistributedCache: sending
↳ServiceConfig ConfigSync to all
2014-09-11 08:47:54.548/13.289 Oracle Coherence GE 12.1.3.0.0 <D5>
↳(thread=Invocation:Management, member=n/a): Service Management left the cluster
2014-09-11 08:47:54.551/13.292 Oracle Coherence GE 12.1.3.0.0 <D5>
↳(thread=Invocation:Management, member=2): Member 1 left service Management with
↳senior member 1
2014-09-11 08:47:54.552/13.293 Oracle Coherence GE 12.1.3.0.0 <D6>
↳(thread=Invocation:Management, member=2): Service Management: sending ServiceConfig
↳ConfigSync to all

```

```

2014-09-11 08:47:54.552/13.293 Oracle Coherence GE 12.1.3.0.0 <D6>
↳(thread=DistributedCache, member=2): Service DistributedCache: sending
↳PartitionConfig ConfigSync to all
2014-09-11 08:47:54.552/13.293 Oracle Coherence GE 12.1.3.0.0 <D6>
↳(thread=DistributedCache, member=3): Service DistributedCache: received
↳ServiceConfig ConfigSync from member 2 containing 2 entries
2014-09-11 08:47:54.552/13.293 Oracle Coherence GE 12.1.3.0.0 <D5>
↳(thread=Invocation:Management, member=3): Member 1 left service Management with
↳senior member 1
2014-09-11 08:47:54.553/13.294 Oracle Coherence GE 12.1.3.0.0 <D6>
↳(thread=Invocation:Management, member=3): Service Management: received
↳ServiceConfig ConfigSync from member 2 containing 1 entries
2014-09-11 08:47:54.557/13.298 Oracle Coherence GE 12.1.3.0.0 <D6>
↳(thread=DistributedCache, member=3): Service DistributedCache: received
↳PartitionConfig ConfigSync from member 2 containing 257 entries
2014-09-11 08:47:54.758/13.499 Oracle Coherence GE 12.1.3.0.0 <D5> (thread=Cluster,
↳member=2): Member(Id=1, Timestamp=2014-09-11 08:47:54.757, Address=127.0.0.1:40100,
↳MachineId=30438, Location=site:DefaultSite,rack:DefaultRack,machine:DefaultMachine,
↳process:1473, Role=DedicatedStorageEnabledMember) left Cluster with senior member 2
2014-09-11 08:47:54.759/13.500 Oracle Coherence GE 12.1.3.0.0 <D6> (thread=Cluster,
↳member=2): TcpRing disconnected from Member(Id=1, Timestamp=2014-09-11 08:47:54.757,
↳Address=127.0.0.1:40100, MachineId=30438, Location=site:DefaultSite,
↳rack:DefaultRack,machine:DefaultMachine,process:1473,
↳Role=DedicatedStorageEnabledMember) to maintain ring
2014-09-11 08:47:54.758/13.499 Oracle Coherence GE 12.1.3.0.0 <D5> (thread=Cluster,
↳member=3): Member(Id=1, Timestamp=2014-09-11 08:47:54.757, Address=127.0.0.1:40100,
↳MachineId=30438, Location=site:DefaultSite,rack:DefaultRack,machine:DefaultMachine,
↳process:1473, Role=DedicatedStorageEnabledMember) left Cluster with senior member 2
2014-09-11 08:47:54.760/13.501 Oracle Coherence GE 12.1.3.0.0 <D6> (thread=Cluster,
↳member=2): TcpRing connected to Member(Id=3, Timestamp=2014-09-11 08:47:47.981,
↳Address=127.0.0.1:40104, MachineId=30438, Location=site:DefaultSite,
↳rack:DefaultRack,machine:DefaultMachine,process:1473,
↳Role=DedicatedExtendProxyMember)
2014-09-11 08:47:54.760/13.501 Oracle Coherence GE 12.1.3.0.0 <D5> (thread=Cluster,
↳member=3): MemberLeft announcement from Member(Id=1, Timestamp=2014-09-11 08:47:54.
↳757, Address=127.0.0.1:40100, MachineId=30438, Location=site:DefaultSite,
↳rack:DefaultRack,machine:DefaultMachine,process:1473,
↳Role=DedicatedStorageEnabledMember)
2014-09-11 08:47:54.761/13.502 Oracle Coherence GE 12.1.3.0.0 <D5> (thread=Cluster,
↳member=2): MemberLeft announcement from Member(Id=1, Timestamp=2014-09-11 08:47:54.
↳757, Address=127.0.0.1:40100, MachineId=30438, Location=site:DefaultSite,
↳rack:DefaultRack,machine:DefaultMachine,process:1473,
↳Role=DedicatedStorageEnabledMember)
2014-09-11 08:47:54.771/13.512 Oracle Coherence GE 12.1.3.0.0 <D5> (thread=Cluster,
↳member=n/a): Service Cluster left the cluster
2014-09-11 08:47:54.773/13.514 Oracle Coherence GE 12.1.3.0.0 <D6>
↳(thread=NameService:TcpAcceptor, member=n/a): Stopped: TcpAcceptor
↳{Name=NameService:TcpAcceptor, State=(SERVICE_STOPPED), ThreadCount=0,
↳Codec=Codec(Format=POF), Serializer=com.tangosol.net.internal.NameServicePofContext,
↳PingInterval=0, PingTimeout=0, RequestTimeout=0, MaxIncomingMessageSize=0,
↳MaxOutgoingMessageSize=0, SocketProvider=com.oracle.common.internal.net.
↳MultiplexedSocketProvider@7a4e2185, LocalAddress=WrapperSocketAddressProvider
↳{Providers=[WrapperSocketAddressProvider{Providers=[[/127.0.0.1:40100]]}},
↳SocketOptions{LingerTimeout=0, KeepAliveEnabled=true, TcpDelayEnabled=false},
↳ListenBacklog=0, BufferPoolIn=BufferPool(BufferSize=2KB, BufferType=DIRECT,
↳Capacity=Unlimited), BufferPoolOut=BufferPool(BufferSize=2KB, BufferType=DIRECT,
↳Capacity=Unlimited)}
2014-09-11 08:47:54.778/13.519 Oracle Coherence GE 12.1.3.0.0 <D5>
↳(thread=DistributedCache, member=n/a): Service DistributedCache left the cluster

```

```

2014-09-11 08:47:54.781/13.522 Oracle Coherence GE 12.1.3.0.0 <D5>
↳(thread=DistributedCache, member=3): Member 2 left service DistributedCache with
↳senior member 2
2014-09-11 08:47:54.781/13.522 Oracle Coherence GE 12.1.3.0.0 <D6>
↳(thread=DistributedCache, member=3): Service DistributedCache: sending
↳ServiceConfig ConfigSync to all
2014-09-11 08:47:54.784/13.525 Oracle Coherence GE 12.1.3.0.0 <D5>
↳(thread=Invocation:Management, member=3): Member 2 left service Management with
↳senior member 2
2014-09-11 08:47:54.784/13.525 Oracle Coherence GE 12.1.3.0.0 <D6>
↳(thread=Invocation:Management, member=3): Service Management: sending ServiceConfig
↳ConfigSync to all
2014-09-11 08:47:54.781/13.522 Oracle Coherence GE 12.1.3.0.0 <D5>
↳(thread=Invocation:Management, member=n/a): Service Management left the cluster
2014-09-11 08:47:54.987/13.728 Oracle Coherence GE 12.1.3.0.0 <D5> (thread=Cluster,
↳member=3): Member(Id=2, Timestamp=2014-09-11 08:47:54.987, Address=127.0.0.1:40102,
↳MachineId=30438, Location=site:DefaultSite, rack:DefaultRack, machine:DefaultMachine,
↳process:1473, Role=DedicatedStorageEnabledMember) left Cluster with senior member 3
2014-09-11 08:47:54.988/13.729 Oracle Coherence GE 12.1.3.0.0 <D6> (thread=Cluster,
↳member=3): TcpRing disconnected from Member(Id=2, Timestamp=2014-09-11 08:47:54.987,
↳Address=127.0.0.1:40102, MachineId=30438, Location=site:DefaultSite,
↳rack:DefaultRack, machine:DefaultMachine, process:1473,
↳Role=DedicatedStorageEnabledMember) to maintain ring
2014-09-11 08:47:54.988/13.729 Oracle Coherence GE 12.1.3.0.0 <D5> (thread=Cluster,
↳member=3): MemberLeft announcement from Member(Id=2, Timestamp=2014-09-11 08:47:54.
↳987, Address=127.0.0.1:40102, MachineId=30438, Location=site:DefaultSite,
↳rack:DefaultRack, machine:DefaultMachine, process:1473,
↳Role=DedicatedStorageEnabledMember)
2014-09-11 08:47:55.002/13.743 Oracle Coherence GE 12.1.3.0.0 <D5> (thread=Cluster,
↳member=n/a): Service Cluster left the cluster
2014-09-11 08:47:55.015/13.756 Oracle Coherence GE 12.1.3.0.0 <D6>
↳(thread=NameService:TcpAcceptor, member=n/a): Stopped: TcpAcceptor
↳{Name=NameService:TcpAcceptor, State=(SERVICE_STOPPED), ThreadCount=0,
↳Codec=Codec(Format=POF), Serializer=com.tangosol.net.internal.NameServicePofContext,
↳PingInterval=0, PingTimeout=0, RequestTimeout=0, MaxIncomingMessageSize=0,
↳MaxOutgoingMessageSize=0, SocketProvider=com.oracle.common.internal.net.
↳MultiplexedSocketProvider@49c53f6a, LocalAddress=WrapperSocketAddressProvider
↳{Providers=[WrapperSocketAddressProvider{Providers=[[/127.0.0.1:40102]]}},
↳SocketOptions{LingerTimeout=0, KeepAliveEnabled=true, TcpDelayEnabled=false},
↳ListenBacklog=0, BufferPoolIn=BufferPool(BufferSize=2KB, BufferType=DIRECT,
↳Capacity=Unlimited), BufferPoolOut=BufferPool(BufferSize=2KB, BufferType=DIRECT,
↳Capacity=Unlimited)}
2014-09-11 08:47:55.017/13.758 Oracle Coherence GE 12.1.3.0.0 <D6>
↳(thread=Proxy:ExtendProxyServerService:TcpAcceptor, member=n/a): Stopped:
↳TcpAcceptor{Name=Proxy:ExtendProxyServerService:TcpAcceptor, State=(SERVICE_
↳STOPPED), ThreadCount=8, HungThreshold=0, TaskTimeout=0, Codec=Codec(Format=POF),
↳Serializer=com.tangosol.io.DefaultSerializer, PingInterval=0, PingTimeout=0,
↳RequestTimeout=0, MaxIncomingMessageSize=0, MaxOutgoingMessageSize=0,
↳SocketProvider=com.oracle.common.internal.net.MultiplexedSocketProvider@75ee2516,
↳LocalAddress=WrapperSocketAddressProvider{Providers=[[/127.0.0.1:40150]]},
↳SocketOptions{LingerTimeout=0, KeepAliveEnabled=true, TcpDelayEnabled=false},
↳ListenBacklog=0, BufferPoolIn=BufferPool(BufferSize=2KB, BufferType=DIRECT,
↳Capacity=Unlimited), BufferPoolOut=BufferPool(BufferSize=2KB, BufferType=DIRECT,
↳Capacity=Unlimited)}
2014-09-11 08:47:55.020/13.761 Oracle Coherence GE 12.1.3.0.0 <D5>
↳(thread=Proxy:ExtendProxyServerService, member=n/a): Service
↳ExtendProxyServerService left the cluster
2014-09-11 08:47:55.023/13.764 Oracle Coherence GE 12.1.3.0.0 <D5>
↳(thread=DistributedCache, member=n/a): Service DistributedCache left the cluster

```

```
2014-09-11 08:47:55.027/13.768 Oracle Coherence GE 12.1.3.0.0 <D5>
↳(thread=Invocation:Management, member=n/a): Service Management left the cluster
2014-09-11 08:47:55.045/13.786 Oracle Coherence GE 12.1.3.0.0 <D5> (thread=Cluster,
↳member=n/a): Service Cluster left the cluster
2014-09-11 08:47:55.048/13.789 Oracle Coherence GE 12.1.3.0.0 <D6>
↳(thread=NameService:TcpAcceptor, member=n/a): Stopped: TcpAcceptor
↳{Name=NameService:TcpAcceptor, State=(SERVICE_STOPPED), ThreadCount=0,
↳Codec=Codec(Format=POF), Serializer=com.tangosol.net.internal.NameServicePofContext,
↳PingInterval=0, PingTimeout=0, RequestTimeout=0, MaxIncomingMessageSize=0,
↳MaxOutgoingMessageSize=0, SocketProvider=com.oracle.common.internal.net.
↳MultiplexedSocketProvider@75ee2516, LocalAddress=WrapperSocketAddressProvider
↳{Providers=[WrapperSocketAddressProvider(Providers=[[/127.0.0.1:40104]])]},
↳SocketOptions{LingerTimeout=0, KeepAliveEnabled=true, TcpDelayEnabled=false},
↳ListenBacklog=0, BufferPoolIn=BufferPool(BufferSize=2KB, BufferType=DIRECT,
↳Capacity=Unlimited), BufferPoolOut=BufferPool(BufferSize=2KB, BufferType=DIRECT,
↳Capacity=Unlimited)}
```

Sep 11, 2014 8:47:55 AM org.littlegrid.impl.DefaultClusterMemberGroup shutdownAll
INFO: ___ Group of 3 cluster member(s) shutdown in 2622ms ___

Making tests faster

Coherence is pretty fast at starting up and shutting down when you compare it to other software like a database or middleware, such as an application server - however, whilst it is fast and littlegrid helps start up an entire small cluster quickly, it gets a bit tedious if you want to **practice Test Driven Development (TDD)** or have lots of automated integration tests that you run locally or on your *Continuous Integration (CI)* server.

What?

Ways of making your tests start faster, how to avoid the repeated start-up and shutdown cluster time when running lots of test classes or a suite of tests and how to run tests in parallel.

Why?

Faster test start-up and faster execution of all your tests makes practicing TDD and CI easier and nicer - besides, no-one likes waiting if there is a faster way!

How?

littlegrid has a number of ways of making things faster, these are:

- A way of establishing your cluster faster, shaving a few seconds off the initial cluster start-up - this is called the fast-start join timeout *Fast-start join timeout*.
- A way of reusing an established cluster across many test classes or a suite of tests *Reusable cluster member group*.
- A way of running tests in parallel *Parallel tests*.

Fast-start join timeout

What?

The fast-start join timeout feature enables Coherence to establish a new cluster more quickly. By default in `dev` mode, Coherence will wait 3 seconds before establishing a new cluster (if no existing cluster is up and running). The `prod` mode setting by default will wait 30 seconds.

Why?

Getting the cluster up and running faster makes iterative development easier, helping facilitate **Test Driven Development (TDD)**. This option can help establish a cluster *ready for running tests* against in approximately 3 seconds (depending upon your machine) - remember this is a running cluster, Coherence is pretty quick starting when you reduce its join-timeout via the fast-start option.

Note: Newer versions of Coherence can use a small time of typically something 100ms, whilst earlier versions of Coherence can only accept a minimum fast-start time of 1000ms.

How? (when you don't have your own Coherence override file)

If you don't have a Coherence override file (typically called something like `tangosol-coherence-override.xml`), then you can simply use the builder API or one of the other mechanisms for configuring littlegrid, see *Controlling and configuring littlegrid*

```
memberGroup = ClusterMemberGroupUtils.newBuilder()
    .setStorageEnabledCount(1)
    .setCacheConfiguration("simple-cache-config.xml")
    .setFastStartJoinTimeoutMilliseconds(100) // Note: older versions of
↳ Coherence need 1000
    .buildAndConfigureForStorageDisabledClient();
```

Without specifying your own Coherence override file, then littlegrid will use its own in-built override file called `littlegrid-fast-start-coherence-override.xml`, it looks like the below:

```
<coherence>
  <cluster-config>
    <multicast-listener>
      <join-timeout-milliseconds system-property="littlegrid.join.timeout.
↳ milliseconds">1500</join-timeout-milliseconds>
    </multicast-listener>
  </cluster-config>
</coherence>
```

When littlegrid uses its own override file and when the fast-start option has been configured (via the Builder or configuration), then you should see the below warning - this is logged to let you know that littlegrid thinks it is controlling the time that Coherence waits before starting the cluster.

```
WARNING: Fast-start join timeout specified. Note: the fast-start Coherence override
↳ file will now be configured to be used
```

How? (if you have your own Coherence override file)

If you're using your own Coherence override file, then in order for the littlegrid fast-start option to work, you will need to include the `join-timeout-milliseconds` XML element within your own override file - additionally if you'd like to use your own system property name (e.g. `jupiter.join.timeout.milliseconds`), then please refer to *Controlling system property names used by littlegrid*.

```
<coherence>
  <cluster-config>
    <multicast-listener>
      <join-timeout-milliseconds system-property="jupiter.join.timeout.
↵milliseconds">1500</join-timeout-milliseconds>
    </multicast-listener>
  </cluster-config>
</coherence>
```

No coding needed (launch cluster for C++, .Net and easy playing)

REPL - try without any Java coding!

littlegrid 2.15 introduced several REPL (Repeat-Evaluate-Print-Loop) consoles and the ability to perform simple scripting with a simple DSL (Domain-Specific Language).

The easiest way to try the new REPL-DSL console is to use the `StorageDisabledClientReplApp`, here's how to launch it - then type `help` to display the help information:

```
java -cp coherence.jar;littlegrid-2.15.2.jar org.littlegrid.app.
StorageDisabledClientReplApp
```

```
INFO: littlegrid 2.15.1 (http://littlegrid.bitbucket.org) - initialised. Builder values: <default file=48, override file=0, 'littlegrid
builder.' environment variables=0, 'LITTLEGRID_BUILDER.' environment variables=0, 'littlegrid.builder.' system properties=1>. Builder to C
oherence system property mapping values: <default file=39, override file=0>
May 16, 2013 10:12:13 PM org.littlegrid.impl.DefaultClusterMemberGroupBuilder buildClusterMembers
INFO: littlegrid 2.15.1 starting - Storage-enabled: 0, Extend proxy: 0, Storage-enabled proxy: 0, JMX: 0, Custom configured: 0
May 16, 2013 10:12:13 PM org.littlegrid.impl.DefaultClusterMemberGroupBuilder buildClusterMembers
INFO: Group of 0 cluster member(s) started in 4ms, member ids are: []
May 16, 2013 10:12:13 PM org.littlegrid.impl.DefaultClusterMemberGroupBuilder buildAndConfigureFor
INFO: System properties set for client/member: <com.sun.management.jmxremote=false, tangosol.coherence.cluster=littlegridCluster, tangosol.c
oherence.distributed.localstorage=false, tangosol.coherence.extend.address=127.0.0.1, tangosol.coherence.extend.enabled=false, tangosol.cohe
rence.localhost=127.0.0.1, tangosol.coherence.localport=22000, tangosol.coherence.log.stdout, tangosol.coherence.log.level=3, tangosol.coher
ence.machine=DefaultMachine, tangosol.coherence.management.none, tangosol.coherence.management.remote=true, tangosol.coherence.rack=DefaultR
ack, tangosol.coherence.role=StorageDisabledClient, tangosol.coherence.site=DefaultSite, tangosol.coherence.tcp.enabled=true, tangosol.cohe
rence.ttl=0, tangosol.coherence.wka=127.0.0.1, tangosol.coherence.wka.port=22000>

lg> help
start storage enabled - starts a storage enabled member in this process
start storage enabled * n - starts the specified number of storage enabled members in this process
start extend proxy n - starts an Extend proxy member with specified port in this process
start jmx monitor - starts a JMX monitor member in this process
members - displays member ids known to this process (note: these are only for this process)
stop member member_id_1 member_id_N - stops the specified cluster member(s) using their member id
shutdown member member_id_1 member_id_N - shuts down the specified cluster member(s) using their id
stop all - stops all cluster member(s)
shutdown all - shuts down all cluster member(s)
bye - exits this application - same as quit
quit - quits this application - same as bye
sleep duration N - sleeps for the specified time in milliseconds, e.g. 1000
sleep until HH:MI:SS - sleeps until the specified time, e.g. 18:01:02
help - displays this help
date - displays the current date and time
# - a comment line, useful when scripting and wanting to comment scripts
site = siteName - sets site name, this will be used by all future started members
rack = rackName - sets rack name, this will be used by all future started members
machine = machineName - sets machine name, this will be used by all future started members
cohql - launches CohQL console
console - launches Coherence console (not for Extend clients)
lg> _
```

Here's how to batch up a series of commands that will then execute one after another - in this case: `sleep until 22:24:02; stop member 2`

The REPL console application is a really great way to experiment with Coherence - you can even invoke the standard Coherence command-line console or CohQL from the REPL console.

The following were added in littlegrid 2.15:

- `org.littlegrid.app.StorageDisabledClientReplApp` - this application is for running commands as a storage-disabled client
- `org.littlegrid.app.ExtendClientReplApp` - this application is for running commands as an Extend clients and requires that an Extend client cache configuration file is specified from the command-line:

```
java -Dlittlegrid.builder.ClientCacheConfiguration=my-extend-cache-config.xml -cp coherence.jar;littlegrid-2.15.2.jar org.littlegrid.app.ExtendClientReplApp
```

- `org.littlegrid.app.BatchCommandApp` - this application is geared up for running commands from a file and is ideal when used with C++ or .Net clients:

```
java -cp coherence.jar;littlegrid-2.15.2.jar org.littlegrid.app.BatchCommandApp commandFile=simple-commands.txt
```

A command file is simply the DSL commands that you wish to execute, for example:

```
start storage enabled

# this is a comment
sleep 5000

start extend proxy 25001

# check which member exist
members

stop member 2

bye ;
```

littlegrid also has a very simple launcher, useful if all settings can be passed from the command-line:

The launcher application could be used if you wanted to spawn an entire ‘mini-cluster’, perhaps from .Net and then connect to it from your .Net code as an Extend client. Alternatively, this technique is useful if you want to connect an application server (such as WebLogic) to a mini-cluster via Extend.

The launcher can be supplied with an optional simple properties file to be used to control the cluster configuration, additionally system properties and environment variables can also be used if required. Below is an example `my-littlegrid.properties`:

```
StorageEnabledCount=2
ExtendProxyCount=1
FastStartJoinTimeoutMilliseconds=1000
```

The properties that can be used match the Builder API without the prefix ‘set’, so these are the same: `newBuilder().setLogLevel(6)` is the same as a property `LogLevel=6`, which is actually the same as `-Dlittlegrid.builder.LogLevel=6`

To start the launcher, simply have Coherence and littlegrid on your class path.

```
java -cp littlegrid-2.15.2.jar;coherence.jar;. org.littlegrid.ClusterMemberGroupApp my-littlegrid.properties
```

```
java -Dlittlegrid.builder.StorageEnabledCount=2 -Dlittlegrid.builder.
↪ExtendProxyCount=1 \
    -Dlittlegrid.builder.FastStartJoinTimeoutMilliseconds=1000 \
    -cp littlegrid-2.15.2.jar;coherence.jar;. org.littlegrid.
↪ClusterMemberGroupApp
```

```
C:\Windows\system32\cmd.exe - java -cp .\littlegrid-2.13.jar;.\coherence-java-3.7.1.0b27797\coherence\lib\coherence.jar; org.littlegrid.ClusterMemberGr...
WellKnownAddressList{Size=1,
  UKA{Address=127.0.0.1, Port=22000}
}
MasterMemberSet<
  ThisMember=Member{Id=3, Timestamp=2012-11-11 19:43:02.075, Address=127.0.0.1:22004, MachineId=30438, Location=site:DefaultSite,rack:DefaultRack,machine:DefaultMachine,process:5300, Role=DedicatedExtendProxyMember}
  OldestMember=Member{Id=1, Timestamp=2012-11-11 19:42:57.884, Address=127.0.0.1:22000, MachineId=30438, Location=site:DefaultSite,rack:DefaultRack,machine:DefaultMachine,process:5300, Role=DedicatedStorageEnabledMember}
  ActualMemberSet=MemberSet{Size=3
    Member{Id=1, Timestamp=2012-11-11 19:42:57.884, Address=127.0.0.1:22000, MachineId=30438, Location=site:DefaultSite,rack:DefaultRack,machine:DefaultMachine,process:5300, Role=DedicatedStorageEnabledMember}
    Member{Id=2, Timestamp=2012-11-11 19:43:00.131, Address=127.0.0.1:22002, MachineId=30438, Location=site:DefaultSite,rack:DefaultRack,machine:DefaultMachine,process:5300, Role=DedicatedStorageEnabledMember}
    Member{Id=3, Timestamp=2012-11-11 19:43:02.075, Address=127.0.0.1:22004, MachineId=30438, Location=site:DefaultSite,rack:DefaultRack,machine:DefaultMachine,process:5300, Role=DedicatedExtendProxyMember}
  }
  MemberId|ServiceVersion|ServiceJoined|MemberState
  1|3.7.1|2012-11-11 19:42:57.884|JOINED
  2|3.7.1|2012-11-11 19:43:00.131|JOINED
  3|3.7.1|2012-11-11 19:43:02.093|JOINED
  RecycleMillis=1200000
  RecycleSet=MemberSet{Size=0
  }
}
TcpRing{Connections=121}
IpMonitor{AddressListSize=0}
2012-11-11 19:43:02.589/6.512 Oracle Coherence GE 3.7.1.0 <Info> <thread=Proxy:TcpProxyService:TcpAcceptor:TcpProcessor, member=3>: The specified local address "127.0.0.1" is a loopback address; clients running on remote machines will not be able to connect to this TcpAcceptor
2012-11-11 19:43:02.594/6.517 Oracle Coherence GE 3.7.1.0 <Info> <thread=Proxy:TcpProxyService:TcpAcceptor, member=3>: TcpAcceptor now listening for connections on 127.0.0.1:23000
Nov 11, 2012 7:43:02 PM org.littlegrid.impl.DefaultClusterMemberGroupBuilder buildClusterMembers
INFO: ___ Group of cluster member(s) started in 6543ms, member ids: [1, 2, 3] ___
Cluster member group launched, press Enter to shutdown...
```

.Net/C++ Extend client tutorial

TODO: CREATE WINDOWS BATCH FILES FOR TUTORIAL

Warning: This is a new page and it is actively being worked on and so the content is likely to change whilst it gets refined and improved - any feedback or comments are welcome :-)

This tutorial is aimed at non-Java developers that need to launch a simple cluster comprising of at least one Extend proxy and at least one storage-enabled member (cache server), thus enabling them to having a fully working 'mini-cluster' for their .Net or C++ Extend client code to connect to.

Assumptions

Before we get started - let's state a few assumptions:

- You have Java installed

(ideally a Java Development Kit - JDK) as this includes a server mode for Java.

See note #1, maybe assume not and that that is something they can add.

- You have a Coherence JAR (littlegrid works with Coherence 3.5.3 or later)
- You have a littlegrid JAR - [download latest stable release](#).

This tutorial uses a sample littlegrid project called *small-sample*, the tutorial aims to get you up and running as quickly as possible.

[link to download ZIP](#)

[link to download JAR](#)

3. So, the files we are interested in just to launch the mini-cluster are:

```
launch-mini-cluster.sh  littlegrid-builder-override.properties  small-sample-datagrid-cluster-1.0-
SNAPSHOT.jar
```

4. Before proceeding further some environment variables need to be set, these are used so that the batch file doesn't need hard-coded paths for Coherence and littlegrid.

```
set COHERENCE_JAR=../coherence.jar set LITTLEGRID_JAR=../littlegrid...version-blah.jar
```

Remember the server flag might not work

Telnet

Maven

Failover testing

These tests demonstrate how littlegrid enables you to perform failover testing, there are two ways a cluster member can leave the cluster, either by shutting down (leaving the cluster in a polite and orderly way) or by stopping a member (the cluster member leaves immediately without notifying the other cluster members).

With these tests, the standard Java Executor framework is used to essentially perform 'deferred' events, this is where the action to be performed is delayed, in this case allowing the execution of an entry processor to start before the stop or shutdown is performed.

The entry processor in this example sleeps for a period that is longer than the deferred event time, so for example if the entry processor sleeps for 5 seconds, but the deferred event to stop the cluster member is performed after 2 seconds, thus making it possible to observe the behaviour of Coherence when performing a stop or shutdown and an 'active' execution of an entry processing is being performed.

```
import com.tangosol.net.CacheFactory;
import com.tangosol.net.DistributedCacheService;
import com.tangosol.net.NamedCache;
import com.tangosol.util.InvocableMap;
import com.tangosol.util.processor.AbstractProcessor;
import org.junit.After;
import org.junit.Before;
import org.junit.Test;
import org.littlegrid.ClusterMemberGroup;
import org.littlegrid.ClusterMemberGroupUtils;

import java.util.concurrent.Executors;
import java.util.concurrent.ScheduledExecutorService;
import java.util.concurrent.TimeUnit;

import static org.junit.Assert.assertTrue;

/**
 * Entry processor failover integration tests - these tests demonstrate the different
 * behaviour when performing a stop or shutdown on a cluster member whilst it is
↳executing
 * an entry processor, the tests also demonstrate how to use the standard Java
↳Executor
 * framework to essentially perform 'deferred' events.
 */
public class EntryProcessorFailoverIntegrationTest {
    private static final int SECONDS_DELAY_BEFORE_DEFERRED_ACTION = 2;
    private static final int SECONDS_TO_SLEEP_IN_ENTRY_PROCESSOR = SECONDS_DELAY_
↳BEFORE_DEFERRED_ACTION + 3;
```

```

private ClusterMemberGroup memberGroup;
private NamedCache cache;

@Before
public void beforeTest() {
    memberGroup = ClusterMemberGroupUtils.newBuilder()
        .setStorageEnabledCount(3)
        .setCacheConfiguration("simple-cache-config.xml")
        .setLogLevel(3)
        .setFastStartJoinTimeoutMilliseconds(1000)
        .buildAndConfigureForStorageDisabledClient();

    cache = CacheFactory.getCache("test");
}

@After
public void afterTest() {
    ClusterMemberGroupUtils.
↳ shutdownCacheFactoryThenClusterMemberGroups(memberGroup);
}

@Test
public void deferredStopAndEntryProcessorFailover() {
    final String key = "abc";
    final int idOfMemberToStop = ((DistributedCacheService) cache.
↳ getCacheService()).getKeyOwner(key).getId();

    final Runnable runnable = new Runnable() {
        @Override
        public void run() {
↳ without
            // With a stop, the Coherence member leaves the cluster immediately,
            // telling the other members - hence it is like it has been killed
            memberGroup.stopMember(idOfMemberToStop);
        }
    };

    final ScheduledExecutorService executorService = Executors.
↳ newScheduledThreadPool(1);
    executorService.schedule(runnable, SECONDS_DELAY_BEFORE_DEFERRED_ACTION,
↳ TimeUnit.SECONDS);

    final Integer idOfMemberThatCompletedProcessing = (Integer) cache.invoke(key,
↳ new SleepyProcessor());

    // The member being stopped whilst executing the entry processor is *not*
↳ expected
    // to be the member that originally started the processing - i.e. the
↳ processing should
    // be completed on the member who originally had the backup
    assertTrue(idOfMemberThatCompletedProcessing != idOfMemberToStop);
}

@Test
public void deferredShutdownAndEntryProcessorFailover() throws
↳ InterruptedException {

```

```

        final String key = "abc";
        final int idOfMemberToShutdown = ((DistributedCacheService) cache.
↪getCacheService()).getKeyOwner(key).getId();

        final Runnable runnable = new Runnable() {
            @Override
            public void run() {
                // With a shutdown, Coherence should let the entry processor finish_
↪on the original
                // member before the member leaves the cluster
                memberGroup.shutdownMember(idOfMemberToShutdown);
            }
        };

        final ScheduledExecutorService executorService = Executors.
↪newScheduledThreadPool(1);
        executorService.schedule(runnable, SECONDS_DELAY_BEFORE_DEFERRED_ACTION,
↪TimeUnit.SECONDS);

        final Integer idOfMemberThatCompletedProcessing = (Integer) cache.invoke(key,
↪new SleepyProcessor());

        // The member being shutdown whilst executing the entry processor is expected
        // to be the member that originally started the processing
        assertTrue(idOfMemberThatCompletedProcessing == idOfMemberToShutdown);
    }

    /**
     * Simple entry processor that sleeps enabling it to be failed over in
     * certain circumstances.
     */
    public static class SleepyProcessor extends AbstractProcessor {
        @Override
        public Object process(final InvocableMap.Entry entry) {
            final int memberId = CacheFactory.ensureCluster().getLocalMember().
↪getId();

            CacheFactory.log("===== About to sleep");

            try {
                TimeUnit.SECONDS.sleep(SECONDS_TO_SLEEP_IN_ENTRY_PROCESSOR);
            } catch (InterruptedException e) {
                throw new IllegalStateException(e);
            }

            CacheFactory.log("===== Finished sleeping - survived! ID of_
↪member that completed processing: "
                + memberId);

            return memberId;
        }
    }
}

```

With the below simple cache configuration:

```

<cache-config>
  <caching-scheme-mapping>

```

```

    <cache-mapping>
      <cache-name>*/</cache-name>
      <scheme-name>simple-distributed-scheme</scheme-name>
    </cache-mapping>
  </caching-scheme-mapping>

  <caching-schemes>
    <distributed-scheme>
      <scheme-name>simple-distributed-scheme</scheme-name>
      <backing-map-scheme>
        <local-scheme/>
      </backing-map-scheme>
      <autostart>>true</autostart>
    </distributed-scheme>
  </caching-schemes>
</cache-config>

```

This first test demonstrates that when a stop is performed, the member originally executing the entry processor stops and failover occurs, with the new (now) owning member performing the execution.

```

Running EntryProcessorFailoverIntegrationTest
23-Apr-2012 22:45:53 org.littlegrid.support.PropertiesUtils loadProperties
INFO: File 'littlegrid/littlegrid-builder-override.properties' found and '4'
↳ properties loaded
23-Apr-2012 22:45:53 org.littlegrid.support.PropertiesUtils loadProperties
INFO: Properties file(s) not found, no properties loaded: 'littlegrid-builder-
↳ override.properties'
23-Apr-2012 22:45:53 org.littlegrid.impl.DefaultClusterMemberGroupBuilder
↳ loadAndSetBuilderKeysAndValues
INFO: Prefixed 'littlegrid.builder.override' environment variables found '0'
23-Apr-2012 22:45:53 org.littlegrid.impl.DefaultClusterMemberGroupBuilder
↳ loadAndSetBuilderKeysAndValues
INFO: Prefixed 'littlegrid.builder.override' system properties found '0'
23-Apr-2012 22:45:53 org.littlegrid.support.PropertiesUtils loadProperties
INFO: Properties file(s) not found, no properties loaded: 'littlegrid-builder-system-
↳ property-mapping-override.properties' 'littlegrid/littlegrid-builder-system-
↳ property-mapping-override.properties'
23-Apr-2012 22:45:53 org.littlegrid.impl.DefaultClusterMemberGroupBuilder
↳ buildClusterMembers
INFO: ___ littlegrid 2.9 starting - Storage-enabled: 3, Extend proxy: 0, Storage-
↳ enabled Extend proxy: 0, Custom configured: 0, JMX monitor: 0 ___
23-Apr-2012 22:45:53 org.littlegrid.impl.DefaultClusterMemberGroup
↳ outputStartAllMessages
INFO: System properties to be set.: {com.sun.management.jmxremote=false, tangosol.
↳ coherence.cacheconfig=coherence/littlegrid-test-cache-config.xml, tangosol.
↳ coherence.cluster=littlegridCluster, tangosol.coherence.distributed.
↳ localstorage=true, tangosol.coherence.extend.address=127.0.0.1, tangosol.coherence.
↳ localhost=127.0.0.1, tangosol.coherence.localport=22000, tangosol.coherence.
↳ log=stdout, tangosol.coherence.log.level=3, tangosol.coherence.management=none,
↳ tangosol.coherence.management.remote=true, tangosol.coherence.override=littlegrid/
↳ littlegrid-fast-start-coherence-override.xml, tangosol.coherence.
↳ role=DedicatedStorageEnabledMember, tangosol.coherence.tcmp.enabled=true, tangosol.
↳ coherence.ttl=0, tangosol.coherence.wka=127.0.0.1, tangosol.coherence.wka.
↳ port=22000}
23-Apr-2012 22:45:53 org.littlegrid.impl.DefaultClusterMemberGroup
↳ outputStartAllMessages
INFO: Max memory: 736MB, current: 736MB, free memory: 728MB
2012-04-23 22:45:54.429/0.792 Oracle Coherence 3.7.1.0 <Info> (thread=pool-1-thread-1,
↳ member=n/a): Loaded operational configuration from "jar:file:/home/jhall/maven/
↳ repository/com/oracle/coherence/3.7.1.0/coherence-3.7.1.0.jar!/tangosol-coherence.

```



```

2012-04-23 22:45:54.432/0.795 Oracle Coherence 3.7.1.0 <Info> (thread=pool-1-thread-1,
↳ member=n/a): Loaded operational overrides from "file:/home/jhall/work/littlegrid/
↳ littlegrid-coherence-testsupport/target/classes/littlegrid/littlegrid-fast-start-
↳ coherence-override.xml"

Oracle Coherence Version 3.7.1.0 Build 27797
Grid Edition: Development mode
Copyright (c) 2000, 2011, Oracle and/or its affiliates. All rights reserved.

2012-04-23 22:45:54.578/0.941 Oracle Coherence GE 3.7.1.0 <Info> (thread=pool-1-
↳ thread-1, member=n/a): Loaded cache configuration from "file:/home/jhall/work/
↳ littlegrid/littlegrid-coherence-testsupport/target/test-classes/coherence/
↳ littlegrid-test-cache-config.xml"; this document does not refer to any schema
↳ definition and has not been validated.
2012-04-23 22:45:55.037/1.400 Oracle Coherence GE 3.7.1.0 <Warning> (thread=pool-1-
↳ thread-1, member=n/a): Local address "127.0.0.1" is a loopback address; this
↳ cluster node will not connect to nodes located on different machines
2012-04-23 22:45:55.224/1.587 Oracle Coherence GE 3.7.1.0 <Warning> (thread=pool-1-
↳ thread-1, member=n/a): PreferredUnicastUdpSocket failed to set receive buffer size
↳ to 511 packets (7.98MB); actual size is 1%, 7 packets (127KB). Consult your OS
↳ documentation regarding increasing the maximum socket buffer size. Proceeding with
↳ the actual value may cause sub-optimal performance.
2012-04-23 22:45:56.919/3.282 Oracle Coherence GE 3.7.1.0 <Info> (thread=Cluster,
↳ member=n/a): Created a new cluster "littlegridCluster" with Member(Id=1,
↳ Timestamp=2012-04-23 22:45:55.267, Address=127.0.0.1:22000, MachineId=60314,
↳ Location=site:,machine:localhost,process:23051, Role=DedicatedStorageEnabledMember,
↳ Edition=Grid Edition, Mode=Development, CpuCount=4, SocketCount=2)
↳ UID=0x7F00000100000136E12B0743EB9A55F0
2012-04-23 22:45:56.924/3.287 Oracle Coherence GE 3.7.1.0 <Info> (thread=pool-1-
↳ thread-1, member=n/a): Started cluster Name=littlegridCluster

WellKnownAddressList(Size=1,
WKA{Address=127.0.0.1, Port=22000}
)

... start-up of Coherence members
2012-04-23 22:46:00.944/7.307 Oracle Coherence GE 3.7.1.0 <Info> (thread=main,
↳ member=n/a): Started cluster Name=littlegridCluster

WellKnownAddressList(Size=1,
WKA{Address=127.0.0.1, Port=22000}
)

MasterMemberSet (
ThisMember=Member(Id=4, Timestamp=2012-04-23 22:46:00.721, Address=127.0.0.1:22006,
↳ MachineId=60314, Location=site:,machine:localhost,process:23051,
↳ Role=StorageDisabledClient)
OldestMember=Member(Id=1, Timestamp=2012-04-23 22:45:55.267, Address=127.0.0.1:22000,
↳ MachineId=60314, Location=site:,machine:localhost,process:23051,
↳ Role=DedicatedStorageEnabledMember)
ActualMemberSet=MemberSet(Size=4
Member(Id=1, Timestamp=2012-04-23 22:45:55.267, Address=127.0.0.1:22000,
↳ MachineId=60314, Location=site:,machine:localhost,process:23051,
↳ Role=DedicatedStorageEnabledMember)
Member(Id=2, Timestamp=2012-04-23 22:45:58.81, Address=127.0.0.1:22002,
↳ MachineId=60314, Location=site:,machine:localhost,process:23051,
↳ Role=DedicatedStorageEnabledMember)
Member(Id=3, Timestamp=2012-04-23 22:45:58.823, Address=127.0.0.1:22004,
↳ MachineId=60314, Location=site:,machine:localhost,process:23051,
↳ Role=DedicatedStorageEnabledMember)

```

```

Member(Id=4, Timestamp=2012-04-23 22:46:00.721, Address=127.0.0.1:22006,
↳MachineId=60314, Location=site:,machine:localhost,process:23051,
↳Role=StorageDisabledClient)
)
MemberId|ServiceVersion|ServiceJoined|MemberState
1|3.7.1|2012-04-23 22:45:55.267|JOINED,
2|3.7.1|2012-04-23 22:45:58.81|JOINED,
3|3.7.1|2012-04-23 22:45:58.823|JOINED,
4|3.7.1|2012-04-23 22:46:00.934|JOINED
RecycleMillis=1200000
RecycleSet=MemberSet(Size=0
)
)

TcpRing{Connections=[3]}
IpMonitor{AddressListSize=0}

2012-04-23 22:46:01.196/7.559 Oracle Coherence GE 3.7.1.0 <Info>
↳(thread=DistributedCache, member=4): Loaded POF configuration from "file:/home/
↳jhall/work/littlegrid/littlegrid-coherence-testsupport/target/test-classes/
↳coherence/littlegrid-test-pof-config.xml"; this document does not refer to any
↳schema definition and has not been validated.
2012-04-23 22:46:01.228/7.591 Oracle Coherence GE 3.7.1.0 <Info>
↳(thread=DistributedCache, member=4): Loaded included POF configuration from
↳"jar:file:/home/jhall/maven/repository/com/oracle/coherence/3.7.1.0/coherence-3.7.1.
↳0.jar!/coherence-pof-config.xml"
2012-04-23 22:46:01.396/7.759 Oracle Coherence GE 3.7.1.0 <Info>
↳(thread=DistributedCache, member=2): ===== About to sleep
23-Apr-2012 22:46:03 org.littlegrid.impl.DefaultClusterMemberGroup stopMember
INFO: About to stop cluster member with id '2'
2012-04-23 22:46:03.486/9.849 Oracle Coherence GE 3.7.1.0 <Info>
↳(thread=DistributedCache, member=1): Restored from backup 128 partitions
2012-04-23 22:46:03.497/9.860 Oracle Coherence GE 3.7.1.0 <Info>
↳(thread=DistributedCache, member=1): ===== About to sleep
2012-04-23 22:46:08.498/14.861 Oracle Coherence GE 3.7.1.0 <Info>
↳(thread=DistributedCache, member=1): ===== Finished sleeping - survived! ID
↳of member that completed processing: 1
23-Apr-2012 22:46:08 org.littlegrid.impl.DefaultClusterMemberGroup shutdownAll
INFO: Restoring system properties back to their original state before member group
↳started
23-Apr-2012 22:46:08 org.littlegrid.impl.DefaultClusterMemberGroup shutdownAll
INFO: Shutting down '3' cluster member(s) in group
2012-04-23 22:46:08.583/14.946 Oracle Coherence GE 3.7.1.0 <Info>
↳(thread=DistributedCache, member=n/a): Remains to transfer before shutting down:
↳129 primary partitions, 128 backup partitions
2012-04-23 22:46:10.585/16.948 Oracle Coherence GE 3.7.1.0 <Info>
↳(thread=DistributedCache, member=n/a): Remains to transfer before shutting down:
↳129 primary partitions, 110 backup partitions
23-Apr-2012 22:46:10 org.littlegrid.impl.DefaultClusterMemberGroup shutdownAll
INFO: ___ Group of cluster member(s) shutdown ___

```

This second test demonstrates that when a shutdown is performed, the member originally executing the entry processor receives the shutdown - but it completes the execution of the entry processor, basically it doesn't leave the cluster until its work is done.

```

23-Apr-2012 22:46:10 org.littlegrid.support.PropertiesUtils loadProperties
INFO: File 'littlegrid/littlegrid-builder-override.properties' found and '4'
↳properties loaded

```

```

23-Apr-2012 22:46:10 org.littlegrid.support.PropertiesUtils loadProperties
INFO: Properties file(s) not found, no properties loaded: 'littlegrid-builder-
↳override.properties'
23-Apr-2012 22:46:10 org.littlegrid.impl.DefaultClusterMemberGroupBuilder_
↳loadAndSetBuilderKeysAndValues
INFO: Prefixed 'littlegrid.builder.override' environment variables found '0'
23-Apr-2012 22:46:10 org.littlegrid.impl.DefaultClusterMemberGroupBuilder_
↳loadAndSetBuilderKeysAndValues
INFO: Prefixed 'littlegrid.builder.override' system properties found '0'
23-Apr-2012 22:46:10 org.littlegrid.support.PropertiesUtils loadProperties
INFO: Properties file(s) not found, no properties loaded: 'littlegrid-builder-system-
↳property-mapping-override.properties' 'littlegrid/littlegrid-builder-system-
↳property-mapping-override.properties'
23-Apr-2012 22:46:10 org.littlegrid.impl.DefaultClusterMemberGroupBuilder_
↳buildClusterMembers
INFO: ___ littlegrid 2.9 starting - Storage-enabled: 3, Extend proxy: 0, Storage-
↳enabled Extend proxy: 0, Custom configured: 0, JMX monitor: 0 ___
23-Apr-2012 22:46:10 org.littlegrid.impl.DefaultClusterMemberGroup_
↳outputStartAllMessages
INFO: System properties to be set.: {com.sun.management.jmxremote=false, tangosol.
↳coherence.cacheconfig=coherence/littlegrid-test-cache-config.xml, tangosol.
↳coherence.cluster=littlegridCluster, tangosol.coherence.distributed.
↳localstorage=true, tangosol.coherence.extend.address=127.0.0.1, tangosol.coherence.
↳localhost=127.0.0.1, tangosol.coherence.localport=22000, tangosol.coherence.
↳log=stdout, tangosol.coherence.log.level=3, tangosol.coherence.management=none,
↳tangosol.coherence.management.remote=true, tangosol.coherence.override=littlegrid/
↳littlegrid-fast-start-coherence-override.xml, tangosol.coherence.
↳role=DedicatedStorageEnabledMember, tangosol.coherence.tcmp.enabled=true, tangosol.
↳coherence.ttl=0, tangosol.coherence.wka=127.0.0.1, tangosol.coherence.wka.
↳port=22000}
23-Apr-2012 22:46:10 org.littlegrid.impl.DefaultClusterMemberGroup_
↳outputStartAllMessages
INFO: Max memory: 736MB, current: 736MB, free memory: 528MB
2012-04-23 22:46:10.995/17.358 Oracle Coherence 3.7.1.0 <Info> (thread=pool-3-thread-
↳1, member=n/a): Loaded operational configuration from "jar:file:/home/jhall/maven/
↳repository/com/oracle/coherence/3.7.1.0/coherence-3.7.1.0.jar!/tangosol-coherence.
↳xml"
2012-04-23 22:46:10.998/17.361 Oracle Coherence 3.7.1.0 <Info> (thread=pool-3-thread-
↳1, member=n/a): Loaded operational overrides from "file:/home/jhall/work/littlegrid/
↳littlegrid-coherence-testsupport/target/classes/littlegrid/littlegrid-fast-start-
↳coherence-override.xml"

Oracle Coherence Version 3.7.1.0 Build 27797
Grid Edition: Development mode
Copyright (c) 2000, 2011, Oracle and/or its affiliates. All rights reserved.

2012-04-23 22:46:11.141/17.504 Oracle Coherence GE 3.7.1.0 <Info> (thread=pool-3-
↳thread-1, member=n/a): Loaded cache configuration from "file:/home/jhall/work/
↳littlegrid/littlegrid-coherence-testsupport/target/test-classes/coherence/
↳littlegrid-test-cache-config.xml"; this document does not refer to any schema_
↳definition and has not been validated.
2012-04-23 22:46:11.420/17.783 Oracle Coherence GE 3.7.1.0 <Warning> (thread=pool-3-
↳thread-1, member=n/a): Local address "127.0.0.1" is a loopback address; this_
↳cluster node will not connect to nodes located on different machines
2012-04-23 22:46:11.444/17.807 Oracle Coherence GE 3.7.1.0 <Warning> (thread=pool-3-
↳thread-1, member=n/a): UnicastUdpSocket failed to set send buffer size to 8 packets_
↳(511KB); actual size is 25%, 2 packets (127KB). Consult your OS documentation_
↳regarding increasing the maximum socket buffer size. Proceeding with the actual_
↳value may cause sub-optimal performance.

```

```

2012-04-23 22:46:11.444/17.807 Oracle Coherence GE 3.7.1.0 <Warning> (thread=pool-3-
↳thread-1, member=n/a): UnicastUdpSocket failed to set receive buffer size to 64
↳packets (3.99MB); actual size is 3%, 2 packets (127KB). Consult your OS
↳documentation regarding increasing the maximum socket buffer size. Proceeding with
↳the actual value may cause sub-optimal performance.
2012-04-23 22:46:11.444/17.807 Oracle Coherence GE 3.7.1.0 <Warning> (thread=pool-3-
↳thread-1, member=n/a): PreferredUnicastUdpSocket failed to set receive buffer size
↳to 511 packets (7.98MB); actual size is 1%, 7 packets (127KB). Consult your OS
↳documentation regarding increasing the maximum socket buffer size. Proceeding with
↳the actual value may cause sub-optimal performance.
2012-04-23 22:46:13.128/19.491 Oracle Coherence GE 3.7.1.0 <Info> (thread=Cluster,
↳member=n/a): Created a new cluster "littlegridCluster" with Member(Id=1,
↳Timestamp=2012-04-23 22:46:11.482, Address=127.0.0.1:22000, MachineId=60314,
↳Location=site:,machine:localhost,process:23051, Role=DedicatedStorageEnabledMember,
↳Edition=Grid Edition, Mode=Development, CpuCount=4, SocketCount=2)
↳UID=0x7F00000100000136E12B469AEB9A55F0
2012-04-23 22:46:13.131/19.494 Oracle Coherence GE 3.7.1.0 <Info> (thread=pool-3-
↳thread-1, member=n/a): Started cluster Name=littlegridCluster

WellKnownAddressList(Size=1,
WKA{Address=127.0.0.1, Port=22000}
)

MasterMemberSet(
ThisMember=Member(Id=1, Timestamp=2012-04-23 22:46:11.482, Address=127.0.0.1:22000,
↳MachineId=60314, Location=site:,machine:localhost,process:23051,
↳Role=DedicatedStorageEnabledMember)
OldestMember=Member(Id=1, Timestamp=2012-04-23 22:46:11.482, Address=127.0.0.1:22000,
↳MachineId=60314, Location=site:,machine:localhost,process:23051,
↳Role=DedicatedStorageEnabledMember)
ActualMemberSet=MemberSet(Size=1
Member(Id=1, Timestamp=2012-04-23 22:46:11.482, Address=127.0.0.1:22000,
↳MachineId=60314, Location=site:,machine:localhost,process:23051,
↳Role=DedicatedStorageEnabledMember)
)
MemberId|ServiceVersion|ServiceJoined|MemberState
1|3.7.1|2012-04-23 22:46:13.128|JOINED
RecycleMillis=1200000
RecycleSet=MemberSet(Size=0
)
)

TcpRing{Connections=[]}
IpMonitor{AddressListSize=0}

2012-04-23 22:46:13.353/19.716 Oracle Coherence GE 3.7.1.0 <Info>
↳(thread=DistributedCache, member=1): Loaded POF configuration from "file:/home/
↳jhall/work/littlegrid/littlegrid-coherence-testsupport/target/test-classes/
↳coherence/littlegrid-test-pof-config.xml"; this document does not refer to any
↳schema definition and has not been validated.
2012-04-23 22:46:13.387/19.750 Oracle Coherence GE 3.7.1.0 <Info>
↳(thread=DistributedCache, member=1): Loaded included POF configuration from
↳"jar:file:/home/jhall/maven/repository/com/oracle/coherence/3.7.1.0/coherence-3.7.1.
↳0.jar!/coherence-pof-config.xml"
2012-04-23 22:46:13.720/20.083 Oracle Coherence 3.7.1.0 <Info> (thread=pool-3-thread-
↳3, member=n/a): Loaded operational configuration from "jar:file:/home/jhall/maven/
↳repository/com/oracle/coherence/3.7.1.0/coherence-3.7.1.0.jar!/tangosol-coherence.
↳xml"

```

```
2012-04-23 22:46:13.724/20.087 Oracle Coherence 3.7.1.0 <Info> (thread=pool-3-thread-3, member=n/a): Loaded operational overrides from "file:/home/jhall/work/littlegrid/littlegrid-coherence-testsupport/target/classes/littlegrid/littlegrid-fast-start-coherence-override.xml"
```

```
Oracle Coherence Version 3.7.1.0 Build 27797
Grid Edition: Development mode
Copyright (c) 2000, 2011, Oracle and/or its affiliates. All rights reserved.
```

```
2012-04-23 22:46:13.839/20.202 Oracle Coherence GE 3.7.1.0 <Info> (thread=pool-3-thread-3, member=n/a): Loaded cache configuration from "file:/home/jhall/work/littlegrid/littlegrid-coherence-testsupport/target/test-classes/coherence/littlegrid-test-cache-config.xml"; this document does not refer to any schema definition and has not been validated.
```

```
2012-04-23 22:46:13.775/20.139 Oracle Coherence 3.7.1.0 <Info> (thread=pool-3-thread-2, member=n/a): Loaded operational configuration from "jar:file:/home/jhall/maven/repository/com/oracle/coherence/3.7.1.0/coherence-3.7.1.0.jar!/tangosol-coherence.xml"
```

```
2012-04-23 22:46:13.779/20.142 Oracle Coherence 3.7.1.0 <Info> (thread=pool-3-thread-2, member=n/a): Loaded operational overrides from "file:/home/jhall/work/littlegrid/littlegrid-coherence-testsupport/target/classes/littlegrid/littlegrid-fast-start-coherence-override.xml"
```

```
Oracle Coherence Version 3.7.1.0 Build 27797
Grid Edition: Development mode
Copyright (c) 2000, 2011, Oracle and/or its affiliates. All rights reserved.
```

```
2012-04-23 22:46:13.895/20.258 Oracle Coherence GE 3.7.1.0 <Info> (thread=pool-3-thread-2, member=n/a): Loaded cache configuration from "file:/home/jhall/work/littlegrid/littlegrid-coherence-testsupport/target/test-classes/coherence/littlegrid-test-cache-config.xml"; this document does not refer to any schema definition and has not been validated.
```

```
2012-04-23 22:46:14.233/20.596 Oracle Coherence GE 3.7.1.0 <Warning> (thread=pool-3-thread-3, member=n/a): Local address "127.0.0.1" is a loopback address; this cluster node will not connect to nodes located on different machines
```

```
2012-04-23 22:46:14.264/20.627 Oracle Coherence GE 3.7.1.0 <Warning> (thread=pool-3-thread-2, member=n/a): Local address "127.0.0.1" is a loopback address; this cluster node will not connect to nodes located on different machines
```

```
2012-04-23 22:46:14.266/20.629 Oracle Coherence GE 3.7.1.0 <Warning> (thread=pool-3-thread-3, member=n/a): UnicastUdpSocket failed to set send buffer size to 8 packets (511KB); actual size is 25%, 2 packets (127KB). Consult your OS documentation regarding increasing the maximum socket buffer size. Proceeding with the actual value may cause sub-optimal performance.
```

```
2012-04-23 22:46:14.266/20.629 Oracle Coherence GE 3.7.1.0 <Warning> (thread=pool-3-thread-3, member=n/a): UnicastUdpSocket failed to set receive buffer size to 64 packets (3.99MB); actual size is 3%, 2 packets (127KB). Consult your OS documentation regarding increasing the maximum socket buffer size. Proceeding with the actual value may cause sub-optimal performance.
```

```
2012-04-23 22:46:14.267/20.630 Oracle Coherence GE 3.7.1.0 <Warning> (thread=pool-3-thread-3, member=n/a): PreferredUnicastUdpSocket failed to set receive buffer size to 511 packets (7.98MB); actual size is 1%, 7 packets (127KB). Consult your OS documentation regarding increasing the maximum socket buffer size. Proceeding with the actual value may cause sub-optimal performance.
```

```
2012-04-23 22:46:14.293/20.656 Oracle Coherence GE 3.7.1.0 <Warning> (thread=pool-3-thread-2, member=n/a): UnicastUdpSocket failed to set send buffer size to 8 packets (511KB); actual size is 25%, 2 packets (127KB). Consult your OS documentation regarding increasing the maximum socket buffer size. Proceeding with the actual value may cause sub-optimal performance.
```

```

2012-04-23 22:46:14.294/20.657 Oracle Coherence GE 3.7.1.0 <Warning> (thread=pool-3-
↳thread-2, member=n/a): UnicastUdpSocket failed to set receive buffer size to 64
↳packets (3.99MB); actual size is 3%, 2 packets (127KB). Consult your OS
↳documentation regarding increasing the maximum socket buffer size. Proceeding with
↳the actual value may cause sub-optimal performance.
2012-04-23 22:46:14.295/20.658 Oracle Coherence GE 3.7.1.0 <Warning> (thread=pool-3-
↳thread-2, member=n/a): PreferredUnicastUdpSocket failed to set receive buffer size
↳to 511 packets (7.98MB); actual size is 1%, 7 packets (127KB). Consult your OS
↳documentation regarding increasing the maximum socket buffer size. Proceeding with
↳the actual value may cause sub-optimal performance.
2012-04-23 22:46:14.575/20.938 Oracle Coherence GE 3.7.1.0 <Info> (thread=Cluster,
↳member=n/a): This Member(Id=2, Timestamp=2012-04-23 22:46:14.391, Address=127.0.0.
↳1:22002, MachineId=60314, Location=site:,machine:localhost,process:23051,
↳Role=DedicatedStorageEnabledMember, Edition=Grid Edition, Mode=Development,
↳CpuCount=4, SocketCount=2) joined cluster "littlegridCluster" with senior
↳Member(Id=1, Timestamp=2012-04-23 22:46:11.482, Address=127.0.0.1:22000,
↳MachineId=60314, Location=site:,machine:localhost,process:23051,
↳Role=DedicatedStorageEnabledMember, Edition=Grid Edition, Mode=Development,
↳CpuCount=4, SocketCount=2)
2012-04-23 22:46:14.608/20.971 Oracle Coherence GE 3.7.1.0 <Info> (thread=pool-3-
↳thread-3, member=n/a): Started cluster Name=littlegridCluster

WellKnownAddressList(Size=1,
WKA{Address=127.0.0.1, Port=22000}
)

MasterMemberSet(
ThisMember=Member(Id=2, Timestamp=2012-04-23 22:46:14.391, Address=127.0.0.1:22002,
↳MachineId=60314, Location=site:,machine:localhost,process:23051,
↳Role=DedicatedStorageEnabledMember)
OldestMember=Member(Id=1, Timestamp=2012-04-23 22:46:11.482, Address=127.0.0.1:22000,
↳MachineId=60314, Location=site:,machine:localhost,process:23051,
↳Role=DedicatedStorageEnabledMember)
ActualMemberSet=MemberSet(Size=2
Member(Id=1, Timestamp=2012-04-23 22:46:11.482, Address=127.0.0.1:22000,
↳MachineId=60314, Location=site:,machine:localhost,process:23051,
↳Role=DedicatedStorageEnabledMember)
Member(Id=2, Timestamp=2012-04-23 22:46:14.391, Address=127.0.0.1:22002,
↳MachineId=60314, Location=site:,machine:localhost,process:23051,
↳Role=DedicatedStorageEnabledMember)
)
MemberId|ServiceVersion|ServiceJoined|MemberState
1|3.7.1|2012-04-23 22:46:11.482|JOINED,
2|3.7.1|2012-04-23 22:46:14.591|JOINED
RecycleMillis=1200000
RecycleSet=MemberSet(Size=0
)
)

TcpRing{Connections=[1]}
IpMonitor{AddressListSize=0}

2012-04-23 22:46:14.619/20.982 Oracle Coherence GE 3.7.1.0 <Info> (thread=Cluster,
↳member=n/a): This Member(Id=3, Timestamp=2012-04-23 22:46:14.43, Address=127.0.0.
↳1:22004, MachineId=60314, Location=site:,machine:localhost,process:23051,
↳Role=DedicatedStorageEnabledMember, Edition=Grid Edition, Mode=Development,
↳CpuCount=4, SocketCount=2) joined cluster "littlegridCluster" with senior
↳Member(Id=1, Timestamp=2012-04-23 22:46:11.482, Address=127.0.0.1:22000,
↳MachineId=60314, Location=site:,machine:localhost,process:23051,
↳Role=DedicatedStorageEnabledMember, Edition=Grid Edition, Mode=Development,
↳CpuCount=4, SocketCount=2)

```

```

2012-04-23 22:46:14.666/21.029 Oracle Coherence GE 3.7.1.0 <Info> (thread=pool-3-
↳thread-2, member=n/a): Started cluster Name=littlegridCluster

WellKnownAddressList (Size=1,
WKA{Address=127.0.0.1, Port=22000}
)

MasterMemberSet (
ThisMember=Member (Id=3, Timestamp=2012-04-23 22:46:14.43, Address=127.0.0.1:22004,
↳MachineId=60314, Location=site:,machine:localhost,process:23051,
↳Role=DedicatedStorageEnabledMember)
OldestMember=Member (Id=1, Timestamp=2012-04-23 22:46:11.482, Address=127.0.0.1:22000,
↳MachineId=60314, Location=site:,machine:localhost,process:23051,
↳Role=DedicatedStorageEnabledMember)
ActualMemberSet=MemberSet (Size=3
Member (Id=1, Timestamp=2012-04-23 22:46:11.482, Address=127.0.0.1:22000,
↳MachineId=60314, Location=site:,machine:localhost,process:23051,
↳Role=DedicatedStorageEnabledMember)
Member (Id=2, Timestamp=2012-04-23 22:46:14.391, Address=127.0.0.1:22002,
↳MachineId=60314, Location=site:,machine:localhost,process:23051,
↳Role=DedicatedStorageEnabledMember)
Member (Id=3, Timestamp=2012-04-23 22:46:14.43, Address=127.0.0.1:22004,
↳MachineId=60314, Location=site:,machine:localhost,process:23051,
↳Role=DedicatedStorageEnabledMember)
)
MemberId|ServiceVersion|ServiceJoined|MemberState
1|3.7.1|2012-04-23 22:46:11.482|JOINED,
2|3.7.1|2012-04-23 22:46:14.391|JOINED,
3|3.7.1|2012-04-23 22:46:14.654|JOINED
RecycleMillis=1200000
RecycleSet=MemberSet (Size=0
)
)

TcpRing{Connections=[2]}
IpMonitor{AddressListSize=0}

2012-04-23 22:46:14.919/21.282 Oracle Coherence GE 3.7.1.0 <Info>
↳(thread=DistributedCache, member=2): Loaded POF configuration from "file:/home/
↳jhall/work/littlegrid/littlegrid-coherence-testsupport/target/test-classes/
↳coherence/littlegrid-test-pof-config.xml"; this document does not refer to any
↳schema definition and has not been validated.
2012-04-23 22:46:14.962/21.325 Oracle Coherence GE 3.7.1.0 <Info>
↳(thread=DistributedCache, member=3): Loaded POF configuration from "file:/home/
↳jhall/work/littlegrid/littlegrid-coherence-testsupport/target/test-classes/
↳coherence/littlegrid-test-pof-config.xml"; this document does not refer to any
↳schema definition and has not been validated.
2012-04-23 22:46:14.978/21.341 Oracle Coherence GE 3.7.1.0 <Info>
↳(thread=DistributedCache, member=2): Loaded included POF configuration from
↳"jar:file:/home/jhall/maven/repository/com/oracle/coherence/3.7.1.0/coherence-3.7.1.
↳0.jar!/coherence-pof-config.xml"
2012-04-23 22:46:15.005/21.368 Oracle Coherence GE 3.7.1.0 <Info>
↳(thread=DistributedCache, member=3): Loaded included POF configuration from
↳"jar:file:/home/jhall/maven/repository/com/oracle/coherence/3.7.1.0/coherence-3.7.1.
↳0.jar!/coherence-pof-config.xml"
23-Apr-2012 22:46:15 org.littlegrid.impl.DefaultClusterMemberGroupBuilder
↳buildClusterMembers
INFO: ____ Group of cluster member(s) started, member Ids: [1, 3, 2] ____

```

```

23-Apr-2012 22:46:15 org.littlegrid.impl.DefaultClusterMemberGroupBuilder
↳buildAndConfigureForStorageDisabledClient
INFO: System properties set for client: {com.sun.management.jmxremote=false, tangosol.
↳coherence.cacheconfig=coherence/littlegrid-test-cache-config.xml, tangosol.
↳coherence.cluster=littlegridCluster, tangosol.coherence.distributed.
↳localstorage=false, tangosol.coherence.extend.address=127.0.0.1, tangosol.coherence.
↳extend.enabled=false, tangosol.coherence.localhost=127.0.0.1, tangosol.coherence.
↳localport=22000, tangosol.coherence.log=stdout, tangosol.coherence.log.level=3,
↳tangosol.coherence.management=none, tangosol.coherence.management.remote=true,
↳tangosol.coherence.override=littlegrid/littlegrid-fast-start-coherence-override.xml,
↳ tangosol.coherence.role=StorageDisabledClient, tangosol.coherence.tcmp.
↳enabled=true, tangosol.coherence.ttl=0, tangosol.coherence.wka=127.0.0.1, tangosol.
↳coherence.wka.port=22000}
2012-04-23 22:46:15.399/21.762 Oracle Coherence GE 3.7.1.0 <Info> (thread=main,
↳member=n/a): Loaded operational configuration from "jar:file:/home/jhall/maven/
↳repository/com/oracle/coherence/3.7.1.0/coherence-3.7.1.0.jar!/tangosol-coherence.
↳xml"
2012-04-23 22:46:15.401/21.764 Oracle Coherence GE 3.7.1.0 <Info> (thread=main,
↳member=n/a): Loaded operational overrides from "file:/home/jhall/work/littlegrid/
↳littlegrid-coherence-testsupport/target/classes/littlegrid/littlegrid-fast-start-
↳coherence-override.xml"

Oracle Coherence Version 3.7.1.0 Build 27797
Grid Edition: Development mode
Copyright (c) 2000, 2011, Oracle and/or its affiliates. All rights reserved.

2012-04-23 22:46:15.407/21.770 Oracle Coherence GE 3.7.1.0 <Info> (thread=main,
↳member=n/a): Loaded cache configuration from "file:/home/jhall/work/littlegrid/
↳littlegrid-coherence-testsupport/target/test-classes/coherence/littlegrid-test-
↳cache-config.xml"; this document does not refer to any schema definition and has
↳not been validated.
2012-04-23 22:46:15.418/21.781 Oracle Coherence GE 3.7.1.0 <Warning> (thread=main,
↳member=n/a): Local address "127.0.0.1" is a loopback address; this cluster node
↳will not connect to nodes located on different machines
2012-04-23 22:46:15.420/21.783 Oracle Coherence GE 3.7.1.0 <Warning> (thread=main,
↳member=n/a): UnicastUdpSocket failed to set send buffer size to 8 packets (511KB);
↳actual size is 25%, 2 packets (127KB). Consult your OS documentation regarding
↳increasing the maximum socket buffer size. Proceeding with the actual value may
↳cause sub-optimal performance.
2012-04-23 22:46:15.420/21.783 Oracle Coherence GE 3.7.1.0 <Warning> (thread=main,
↳member=n/a): UnicastUdpSocket failed to set receive buffer size to 64 packets (3.
↳99MB); actual size is 3%, 2 packets (127KB). Consult your OS documentation
↳regarding increasing the maximum socket buffer size. Proceeding with the actual
↳value may cause sub-optimal performance.
2012-04-23 22:46:15.420/21.783 Oracle Coherence GE 3.7.1.0 <Warning> (thread=main,
↳member=n/a): PreferredUnicastUdpSocket failed to set receive buffer size to 511
↳packets (7.98MB); actual size is 1%, 7 packets (127KB). Consult your OS
↳documentation regarding increasing the maximum socket buffer size. Proceeding with
↳the actual value may cause sub-optimal performance.
2012-04-23 22:46:15.602/21.965 Oracle Coherence GE 3.7.1.0 <Info> (thread=Cluster,
↳member=n/a): Failed to satisfy the variance: allowed=16, actual=47
2012-04-23 22:46:15.602/21.965 Oracle Coherence GE 3.7.1.0 <Info> (thread=Cluster,
↳member=n/a): Increasing allowable variance to 19
2012-04-23 22:46:15.960/22.323 Oracle Coherence GE 3.7.1.0 <Info> (thread=Cluster,
↳member=n/a): This Member(Id=4, Timestamp=2012-04-23 22:46:15.757, Address=127.0.0.
↳1:22006, MachineId=60314, Location=site:,machine:localhost,process:23051,
↳Role=StorageDisabledClient, Edition=Grid Edition, Mode=Development, CpuCount=4,
↳SocketCount=2) joined cluster "littlegridCluster" with senior Member(Id=1,
↳Timestamp=2012-04-23 22:46:11.482, Address=127.0.0.1:22000, MachineId=60314,
↳Location=site:,machine:localhost,process:23051, Role=DedicatedStorageEnabledMember,
↳Edition=Grid Edition, Mode=Development, CpuCount=4, SocketCount=2)

```



```

2012-04-23 22:46:16.150/22.513 Oracle Coherence GE 3.7.1.0 <Info> (thread=main,
↳member=n/a): Started cluster Name=littlegridCluster

WellKnownAddressList(Size=1,
WKA{Address=127.0.0.1, Port=22000}
)

MasterMemberSet(
ThisMember=Member(Id=4, Timestamp=2012-04-23 22:46:15.757, Address=127.0.0.1:22006,
↳MachineId=60314, Location=site:,machine:localhost,process:23051,
↳Role=StorageDisabledClient)
OldestMember=Member(Id=1, Timestamp=2012-04-23 22:46:11.482, Address=127.0.0.1:22000,
↳MachineId=60314, Location=site:,machine:localhost,process:23051,
↳Role=DedicatedStorageEnabledMember)
ActualMemberSet=MemberSet(Size=4
Member(Id=1, Timestamp=2012-04-23 22:46:11.482, Address=127.0.0.1:22000,
↳MachineId=60314, Location=site:,machine:localhost,process:23051,
↳Role=DedicatedStorageEnabledMember)
Member(Id=2, Timestamp=2012-04-23 22:46:14.391, Address=127.0.0.1:22002,
↳MachineId=60314, Location=site:,machine:localhost,process:23051,
↳Role=DedicatedStorageEnabledMember)
Member(Id=3, Timestamp=2012-04-23 22:46:14.43, Address=127.0.0.1:22004,
↳MachineId=60314, Location=site:,machine:localhost,process:23051,
↳Role=DedicatedStorageEnabledMember)
Member(Id=4, Timestamp=2012-04-23 22:46:15.757, Address=127.0.0.1:22006,
↳MachineId=60314, Location=site:,machine:localhost,process:23051,
↳Role=StorageDisabledClient)
)
MemberId|ServiceVersion|ServiceJoined|MemberState
1|3.7.1|2012-04-23 22:46:11.482|JOINED,
2|3.7.1|2012-04-23 22:46:14.391|JOINED,
3|3.7.1|2012-04-23 22:46:14.43|JOINED,
4|3.7.1|2012-04-23 22:46:16.136|JOINED
RecycleMillis=1200000
RecycleSet=MemberSet(Size=0
)
)

TcpRing{Connections=[3]}
IpMonitor{AddressListSize=0}

2012-04-23 22:46:16.283/22.646 Oracle Coherence GE 3.7.1.0 <Info>
↳(thread=DistributedCache, member=2): ===== About to sleep
23-Apr-2012 22:46:18 org.littlegrid.impl.DefaultClusterMemberGroup shutdownMember
INFO: About to shutdown cluster member '2'
2012-04-23 22:46:21.284/27.647 Oracle Coherence GE 3.7.1.0 <Info>
↳(thread=DistributedCache, member=n/a): ===== Finished sleeping - survived!
↳ID of member that completed processing: 2
2012-04-23 22:46:21.289/27.652 Oracle Coherence GE 3.7.1.0 <Info>
↳(thread=DistributedCache, member=n/a): Remains to transfer before shutting down: 86
↳primary partitions, 85 backup partitions
23-Apr-2012 22:46:21 org.littlegrid.impl.DefaultClusterMemberGroup shutdownAll
INFO: Restoring system properties back to their original state before member group
↳started
23-Apr-2012 22:46:21 org.littlegrid.impl.DefaultClusterMemberGroup shutdownAll
INFO: Shutting down '3' cluster member(s) in group
2012-04-23 22:46:21.382/27.745 Oracle Coherence GE 3.7.1.0 <Info>
↳(thread=DistributedCache, member=n/a): Remains to transfer before shutting down: 85
↳primary partitions, 0 backup partitions

```

```

2012-04-23 22:46:21.412/27.775 Oracle Coherence GE 3.7.1.0 <Info>
↳(thread=DistributedCache, member=n/a): Remains to transfer before shutting down: 85
↳primary partitions, 0 backup partitions
2012-04-23 22:46:21.594/27.957 Oracle Coherence GE 3.7.1.0 <Info>
↳(thread=DistributedCache, member=n/a): Remains to transfer before shutting down: 1
↳primary partitions, 0 backup partitions
2012-04-23 22:46:21.751/28.115 Oracle Coherence GE 3.7.1.0 <Info>
↳(thread=DistributedCache, member=n/a): Remains to transfer before shutting down: 1
↳primary partitions, 0 backup partitions
23-Apr-2012 22:46:22 org.littlegrid.impl.DefaultClusterMemberGroup shutdownAll
INFO: ___ Group of cluster member(s) shutdown ___

```

Multiple autonomous clusters

If you haven't already had a look, the *Getting started with littlegrid* is the best place to start looking at littlegrid.

The below code example show how to run two autonomous clusters within the same JVM using littlegrid.

```

import com.tangosol.net.CacheFactory;
import org.junit.Test;
import org.littlegrid.ClusterMemberGroup;
import org.littlegrid.ClusterMemberGroupUtils;

import static org.junit.Assert.assertEquals;

public class AutonomousClusterIntegrationTest {
    @Test
    public void twoAutonomousClusters() {
        final int numberOfMembers = 2;
        final int expectedClusterSize = numberOfMembers + 1; // Include this test
↳which will join as a member

        ClusterMemberGroup memberGroupCluster1 = null;
        ClusterMemberGroup memberGroupCluster2 = null;

        try {
↳newBuilder()
            .setStorageEnabledCount(numberOfMembers);

            // Build the first cluster (we won't actually be connecting to this one)
            memberGroupCluster1 = builder.buildAndConfigureForNoClient();

            // The second cluster will need to run on a different port to avoid
↳clustering with
            // the first cluster
            final int member2WkaPort = builder.getWkaPort() + 100;

            // Build the second cluster - we will join this cluster through this test
↳by
            // asserting the cluster size
            memberGroupCluster2 = ClusterMemberGroupUtils.newBuilder()
                .setStorageEnabledCount(numberOfMembers)
                .setWkaPort(member2WkaPort)
                .buildAndConfigureForStorageDisabledClient();

```

```

        // Check the size of the second cluster
        assertEquals(expectedClusterSize, CacheFactory.ensureCluster().
↪getMemberSet().size());
    } finally {
        ClusterMemberGroupUtils.shutdownCacheFactoryThenClusterMemberGroups(
            memberGroupCluster1, memberGroupCluster2);
    }
}
}

```

Maven best practice

littlegrid is available from Maven central, simply add the following dependency to your POM file - littlegrid should be set to **test** scope.

Note: littlegrid doesn't have any dependencies on other frameworks, it simply requires a version of Coherence to be available in your project.

```

<dependency>
  <groupId>org.littlegrid</groupId>
  <artifactId>littlegrid</artifactId>
  <version>2.15.2</version>
  <scope>test</scope>
</dependency>

```

littlegrid is used on a lot of Maven projects and so the below is a quick list of best practice to help you get going easily and quickly

1. Ensure littlegrid is set to *test* scope.
2. Create a separate Maven module for the functional/integration tests (basically those using littlegrid to start a Coherence cluster). This module will typically have your tests for things like: entry processors, cache stores, triggers, backing map access etc.

Typical names for this module might be something like:
my-project-datagrid-cluster-integration-tests

3. Configure Surefire to allocate more memory to your functional/integration tests and allow for class-reloading (littlegrid loads classes into separate class-loaders for each Coherence cluster members).

Note: this Surefire configuration is only required for your littlegrid functional/integration test module(s).

```

<build>
  <plugins>
    <plugin>
      <groupId>org.apache.maven.plugins</groupId>
      <artifactId>maven-surefire-plugin</artifactId>
      <version>${maven-surefire-plugin.version}</version>
      <configuration>
        <!--
          Set the forkMode to 'once' if you have many tests and you intend
          to use the reusable cluster member group functionality that
          littlegrid can provide.
        -->

```

```

-->
<forkMode>once</forkMode>

<!--
  Set the forkMode to 'always' if you're just getting started -
  after you've got littlegrid embedded into your build, then look
  to change this to 'once' and also use the littlegrid
  reusable cluster member group functionality.
-->
<!-- <forkMode>always</forkMode> -->
<argLine>-Xmx768m -Xms768m -XX:MaxPermSize=384m</argLine>
</configuration>
</plugin>

...
</plugins>
</build>

```

4. Use a common way to start and shutdown littlegrid, typically littlegrid and your Coherence cluster controlled by littlegrid are started within something like JUnit's @BeforeClass and typically shutdown in an @AfterClass

Note: Often a base class is used which performs the starting up and shutting down of littlegrid, typical examples might be a base class for storage-disabled client tests using one or two storage-enabled cache servers. Other examples are an Extend client, connecting to an Extend proxy and one or more cache servers.

5. Most projects can utilise littlegrid's reusable cluster member group - this basically tells littlegrid to try and use the same established cluster for all your tests, this *Reusable cluster member group* avoids the delay of starting up and shutting down the Coherence cluster members for each test class
6. If your project also uses Extend, then it is recommended that a separate Maven module is introduced just for the Extend tests.

Typical names for this module might be something like:
 my-project-datagrid-remote-client-integration-tests

Again, configure Surefire for this module to use specific memory settings (see above) and also to use the reusable cluster member group.

So, here's how the project might look:

my-project-datagrid-cluster

Actual code for entry processors, cache stores etc. Any standard unit tests (i.e. not integration tests).

my-project-datagrid-cluster-integration-tests

littlegrid integration tests for entry processors, cache stores etc. typically as storage-disabled client and using one or more storage-enabled members.

my-project-datagrid-remote-client

Actual code for Extend remote client, perhaps Spring configuration/wrapper around Coherence etc.

my-project-datagrid-remote-client-integration-tests

littlegrid integration tests connecting as an Extend client with one or more storage-enabled members and one or more Extend proxies.

Continuous Integration (CI)

littlegrid was developed using CI for CI, therefore it is very CI friendly. The technique of running a cluster in a single process (as used by littlegrid) avoids many of the problems associated with running a separate process for each cluster member, such as needing to clear up all the processes if the build breaks or needs to be stopped.

Walking the walking...

Here's the new littlegrid CI server (the old server died) - it is configured with 3 individual build agents and runs on a quad-core Xenon server to ensure that littlegrid works in a highly concurrent build environment.

Note: each commit to the littlegrid Git repository triggers a full build and tests with littlegrid and Coherence 3.5.x, 3.6.x, 3.7.1.x and 12.1.2.x.

The screenshot shows the Hudson web interface in a Google Chrome browser. The main content area is titled "Jobs Status" and displays a table of build jobs. The table has columns for status (S/W), job name, last success/failure, last duration, console, and line coverage. The jobs listed are for various versions of littlegrid-master-coherence and a sample project. The left sidebar contains navigation links like "New Job", "Manage Hudson", "People", "Build History", "New View", and "Disk usage". Below these are sections for "Build Queue" and "Build Executor Status".

S	W	Job	Last Success	Last Failure	Last Duration	Console	Line Coverage
🟢	☀️	littlegrid-master-coherence-3.5.3	11 hr (#81)	N/A	29 min	📄	N/A%
🟢	☀️	littlegrid-master-coherence-3.6.1.8	11 hr (#117)	N/A	8 min 19 sec	📄	N/A%
🟢	☀️	littlegrid-master-coherence-3.7.1.0	11 hr (#111)	N/A	8 min 11 sec	📄	N/A%
🟢	☀️	littlegrid-master-coherence-3.7.1.0-reporting	10 hr (#89)	N/A	9 min 1 sec	📄	N/A%
🟢	☀️	littlegrid-master-coherence-3.7.1.8	10 hr (#66)	N/A	8 min 38 sec	📄	N/A%
🟢	☀️	littlegrid-sample-project	1 mo 4 days (#21)	N/A	50 sec	📄	N/A%

How to get started with littlegrid and CI?

How do I get started - what do I need to do or configure?

Single build plan or **single build agent** servicing many build plans sequentially using littlegrid and Coherence on your CI server or CI agent?

Cool, you don't need to do anything - littlegrid and Coherence will work just fine with their defaults.

Multiple concurrent builds using littlegrid and Coherence on your CI server or CI agent?

If you have multiple concurrent build plans running on your CI server or CI agent at once then you need to configure littlegrid and essentially Coherence to use different WKA and Extend (if required) ports to

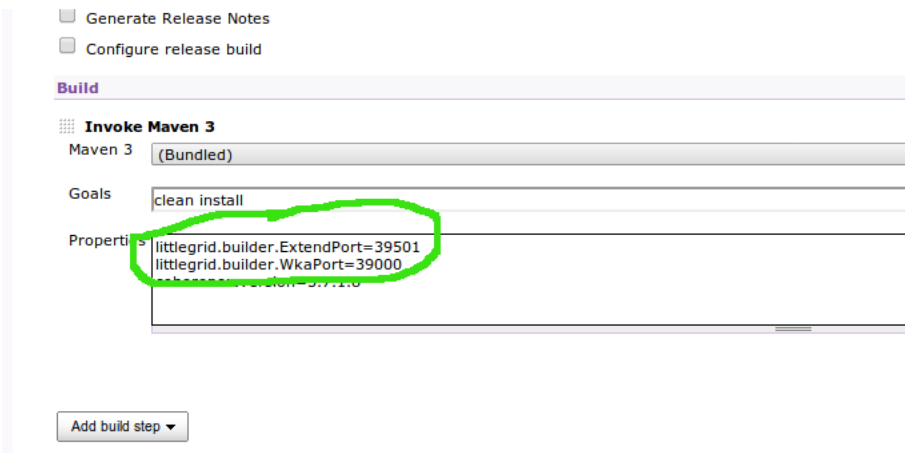
avoid the various build plans all clustering together or the Extend ports being already being used - don't worry, it is really straight forward. The simplest way is to just tell the build agent or CI server to use an Environment variable or System property - here's an example of using a system property:

```
-Dlittlegrid.builder.WkaPort=35000
-Dlittlegrid.builder.ExtendPort=36000
```

If you prefer to use environment variables, then these can be configured like below (in this example for Windows - note the use of underscores):

```
set littlegrid_builder_WkaPort=35000
set littlegrid_builder_ExtendPort=36000
```

Here's what it looks like when configured on Hudson:



More information is available on the [Controlling and configuring littlegrid](#)

That's it, now the build plan that uses these port numbers will avoid using the default littlegrid port numbers for clustering and running Extend.

Controlling and configuring littlegrid

littlegrid can be configured/controlled using a variety of methods, such as:

- [The Builder API](#)
- Properties file
- System properties
- Environment variables

The order of configuration priority is as below, so - any configuration you specify in code is used in preference to any other of the configurations mechanisms, with the littlegrid builder default properties file being the lowest priority:

1. Your code, so for instance:

```
memberGroup = ClusterMemberGroupUtils.newBuilder()
    .setLogLevel(9)
```

2. System properties prefixed with `littlegrid.builder`, such as:

```
-Dlittlegrid.builder.LogLevel=6
```

3. Environment variables prefixed with `littlegrid_builder`, e.g. for Windows:

```
set littlegrid_builder.LogLevel=6
```

or Linux (Bash):

```
export littlegrid_builder.LogLevel=6
```

4. Your optional `littlegrid-builder-override.properties` (loaded from the root of a class-path or from a directory called `littlegrid` below the root of the class-path):

```
littlegrid/littlegrid-builder-override.properties
```

or:

```
littlegrid-builder-override.properties
```

5. The built-in `littlegrid-builder-default.properties` (loaded from the `littlegrid` JAR file) - this contains all the default properties used to configure `littlegrid`.

Note: Whatever mechanism you use to configure `littlegrid`, it essentially keeps an internal mapping of its *builder name* to the specified value.

The below are typically default properties that are used to configure `littlegrid`, any of which can be overridden using the above described methods:

littlegrid builder name	Default value
ExceptionReporterInstanceClassName	org.littlegrid.impl.DefaultBuildExceptionReporter
CallbackHandlerInstanceClassName	org.littlegrid.impl.DefaultCallbackHandler
ClusterMemberInstanceClassName	org.littlegrid.impl.DefaultClusterMember
CustomConfiguredClusterMemberInstanceClassName	org.littlegrid.impl.DefaultClusterMember
NumberOfThreadsInStartUpPool	4
CacheConfiguration	
OverrideConfiguration	
ClientCacheConfiguration	
ClientOverrideConfiguration	
CustomConfiguredCacheConfiguration	
FastStartJoinTimeoutMilliseconds	0
SuggestedSleepAfterStopDuration35x	120
SuggestedSleepAfterStopDuration36x	3
SuggestedSleepAfterStopDurationDefault	3
JarsToExcludeFromClassPath	
CoreJarsToExcludeFromClassPath	alt-rt.jar,charsets.jar,deploy.jar,javaws.jar,jce.jar,jfr.jar,jsse.jar, management-
CustomConfiguredCount	0
StorageEnabledCount	0
StorageEnabledExtendProxyCount	0
ExtendProxyCount	0
JmxMonitorCount	0
ClusterName	littlegridCluster
SiteName	DefaultSite

Table 1.1 – continued

littlegrid builder name	Default value
RackName	DefaultRack
MachineName	DefaultMachine
CustomConfiguredRoleName	CustomConfiguredMember
StorageEnabledRoleName	DedicatedStorageEnabledMember
StorageEnabledExtendProxyRoleName	StorageEnabledExtendProxyMember
ExtendProxyRoleName	DedicatedExtendProxyMember
JmxMonitorRoleName	DedicatedJmxMonitorMember
StorageDisabledClientRoleName	StorageDisabledClient
ExtendClientRoleName	ExtendClient
WkaPort	22000
WkaAddress	127.0.0.1
Ttl	0
ExtendPort	23000
LogDestination	stdout
LogLevel	3

Controlling system property names used by littlegrid

Coherence provides the ability for you to supply default values in your configuration files (such as cache configuration and override file) and for those static values (as defined in the file) to be selectively overridden at runtime, so for instance the address and port that an Extend proxy server listens on might look something like the below:

```
<tcp-acceptor>
  <local-address>
    <address system-property="tangosol.coherence.extend.address">localhost</
    <address>
      <port system-property="tangosol.coherence.extend.port">9099</port>
    </local-address>
  </tcp-acceptor>
```

So, in the above, the default Extend port will be 9099, unless a different values is supplied using a Java system property, such as:

```
java -Dtangosol.coherence.extend.port=20123 -cp some-classpath-etc
```

Note: Most Coherence projects use the ability to override values, so they can have the minimum number of configuration files as possible and avoid having specific configuration files per environment.

What?

littlegrid has a fluent API that is based around the builder pattern, the littlegrid `ClusterMemberGroup.Builder` allows a value to be set and for setting of values to be chained, e.g.

```
memberGroup = ClusterMemberGroupUtils.newBuilder()
    .setWkaPort(15001)
    .setExtendPort(16001)
    ....
```


The values that are set are then mapped to internal names and littlegrid provides a mechanism to then map its internal names to Coherence system property names.

Note: You can configure littlegrid in a number of ways (Builder, properties file, system properties or environment variables), they all internally map to the builder names, see *Controlling and configuring littlegrid*

Why?

Put simply - each project is different! Projects often tend to adopt their own conventions and standard naming.

....so, littlegrid provides the ability to map its internal builder names to Coherence system property names to accommodate different project naming conventions or standards, so for instance if the project is called something like Jupiter, then sometimes the project team may choose to prefix their system properties, e.g.

```
<address system-property="jupiter.extend.port">localhost</address>
```

How?

littlegrid maintains a mapping of builder keys to system properties - this allows it to be easily configured to use custom system properties instead of the standard Coherence system properties. To provide your own mapping, simply include a file called `littlegrid-builder-system-property-mapping-override.properties` on your **class-path in the root directory** or `/littlegrid` directory off the class-path root.

Below is a list of littlegrid builder name to default system property names:

littlegrid builder name	Default system property name
FastStartJoinTimeoutMilliseconds	littlegrid.join.timeout.milliseconds
CacheConfiguration	tangosol.coherence.cacheconfig
ClientCacheConfiguration	tangosol.coherence.cacheconfig
OverrideConfiguration	tangosol.coherence.override
ClientOverrideConfiguration	tangosol.coherence.override
CustomConfiguredCacheConfiguration	tangosol.coherence.cacheconfig
DistributedLocalStorage	tangosol.coherence.distributed.localstorage
TcmpEnabled	tangosol.coherence.tcmp.enabled
ExtendEnabled	tangosol.coherence.extend.enabled
ClusterName	tangosol.coherence.cluster
SiteName	tangosol.coherence.site
RackName	tangosol.coherence.rack
MachineName	tangosol.coherence.machine
CustomConfiguredRoleName	tangosol.coherence.role
StorageEnabledRoleName	tangosol.coherence.role
StorageDisabledClientRoleName	tangosol.coherence.role
StorageEnabledExtendProxyRoleName	tangosol.coherence.role
JmxMonitorRoleName	tangosol.coherence.role
ExtendProxyRoleName	tangosol.coherence.role
ExtendClientRoleName	tangosol.coherence.role
LocalAddress	tangosol.coherence.localhost
LocalPort	tangosol.coherence.localport
WkaAddress	tangosol.coherence.wka
Continued on next page	

Table 1.2 – continued from previous page

littlegrid builder name	Default system property name
WkaPort	tangosol.coherence.wka.port
Ttl	tangosol.coherence.ttl
ExtendAddress	tangosol.coherence.extend.address
ExtendPort	tangosol.coherence.extend.port
LogLevel	tangosol.coherence.log.level
LogDestination	tangosol.coherence.log
CoherenceManagement	tangosol.coherence.management
CoherenceManagementRemote	tangosol.coherence.management.remote
ManagementJmxRemote	com.sun.management.jmxremote

Reusable cluster member group

Typically on a big project it is useful to start a cluster and then reuse it across the various test classes or suites, littlegrid 2.15 introduced a ‘productised’ feature to easily keep an established cluster up and running across test classes (previously there were some workarounds).

SimpleKeepAliveClusterMemberGroup

The simple keep-alive cluster member group enables the cluster member group (essentially members within a cluster) to be reused across test classes. Once created and started, any requests for the same member group return the original instance and when it is requested to be shutdown, it simply ignores the request.

The simple keep-alive cluster member group is ideal for a testing classes within a module that all require the same type of cluster across all the tests. The member group is effectively cleared up when the test process ends.

```
memberGroup = ClusterMemberGroupUtils.newBuilder()
    .setClusterMemberGroupInstanceClassName(SimpleKeepAliveClusterMemberGroup.class.
↳getName())
    .setStorageEnabledCount(2)
    .buildAndConfigureForStorageDisabledClient();

NamedCache cache = CacheFactory.getCache("test");
```

That’s it, simple *Controlling and configuring littlegrid* configure the littlegrid builder ClusterMemberGroupInstanceClassName with the name of the SimpleKeepAliveClusterMemberGroup.

UsageCountingClusterMemberGroup

The usage counting cluster member group is typically useful when test classes have been gathered up into test suites, this technique is typically used on modules where different types of cluster are required across different test classes within the module. Many projects won’t need the usage counting member group and the simple keep-alive member group should be the starting point.

More content will be added soon.

Parallel tests

The below shows a snippet of a POM, a proper description will be added in the future. For now though, the below demonstrates how to configure Surefire and control the number of threads and port numbers for those comfortable with Maven.

What?

Why?

How?

```
<build>
  <plugins>
    <plugin>
      <groupId>org.apache.maven.plugins</groupId>
      <artifactId>maven-surefire-plugin</artifactId>
      <version>${maven-surefire-plugin.version}</version>
      <configuration>
        <forkCount>${fork.count}</forkCount>
        <reuseForks>>false</reuseForks>
        <argLine>-Xmx768m -Xms768m -XX:MaxPermSize=384m</argLine>
        <systemPropertyVariables>
          <littlegrid.builder.WkaPort>${starting.thousand.port.number}$
↪{surefire.forkNumber}00
          </littlegrid.builder.WkaPort>
          <littlegrid.builder.ExtendPort>
            ${starting.thousand.port.number}${surefire.forkNumber}50
          </littlegrid.builder.ExtendPort>
          <littlegrid.builder.ClusterName>littlegridCluster-${surefire.
↪forkNumber}
          </littlegrid.builder.ClusterName>
        </systemPropertyVariables>
      </configuration>
    </plugin>
    ...
  </plugins>
</build>

<properties>
  <starting.thousand.port.number>40</starting.thousand.port.number>
  <fork.count>4</fork.count>
</properties>
```

Sample exception report

The below is a sample exception report that is produced when littlegrid can't start your cluster correctly (in this example, the Coherence JAR wasn't on the class path), in addition to the exception report, littlegrid has attempted to diagnose what it thinks the problem is:

```
Possible exception reason: CHECK_CHILD_FIRST_CLASS_PATH_IN_USE There were 2
member ids [1, 1] - however only these were unique member ids [1]. Ensure that
the Coherence JAR is on your test class path
```

SEVERE: Failed to start cluster member group - please check the exception report to
↳aid diagnosis

Build exception

If you would like help, **then** please email this entire exception report to: ``help at
↳littlegrid dot org``

Exception occurred, trouble-shooting information below:

Name.....: littlegrid
Version number.....: 2.13
Build date.....: Thu Nov 08 21:28:36 GMT 2012
Current time.....: Mon Nov 12 07:56:20 GMT 2012

Possible exception reason: CHECK_CHILD_FIRST_CLASS_PATH_IN_USE There were 2 member
↳ids [1, 1] - however only these were unique member ids [1]. Ensure that the
↳Coherence JAR is on your test class path

Java home.....: /opt/java/jdk1.7.0_04/jre

Class path.....: /opt/intellij/idea-IU-117.117/lib/idea_rt.jar:/opt/
↳intellij/idea-IU-117.117/plugins/junit/lib/junit-rt.jar:/opt/java/jdk1.7.0_04/jre/
↳lib/deploy.jar:/opt/java/jdk1.7.0_04/jre/lib/javaws.jar:/opt/java/jdk1.7.0_04/jre/
↳lib/management-agent.jar:/opt/java/jdk1.7.0_04/jre/lib/charsets.jar:/opt/java/jdk1.
↳7.0_04/jre/lib/jsse.jar:/opt/java/jdk1.7.0_04/jre/lib/jce.jar:/opt/java/jdk1.7.0_04/
↳jre/lib/jfr.jar:/opt/java/jdk1.7.0_04/jre/lib/plugin.jar:/opt/java/jdk1.7.0_04/jre/
↳lib/rt.jar:/opt/java/jdk1.7.0_04/jre/lib/resources.jar:/opt/java/jdk1.7.0_04/jre/
↳lib/ext/sunjce_provider.jar:/opt/java/jdk1.7.0_04/jre/lib/ext/sunpkcs11.jar:/opt/
↳java/jdk1.7.0_04/jre/lib/ext/zipfs.jar:/opt/java/jdk1.7.0_04/jre/lib/ext/dnsns.jar:/
↳opt/java/jdk1.7.0_04/jre/lib/ext/localedata.jar:/opt/java/jdk1.7.0_04/jre/lib/ext/
↳sunec.jar:/home/jhall/work/littlegrid/littlegrid-coherence-testsupport/target/test-
↳classes:/home/jhall/work/littlegrid/littlegrid-coherence-testsupport/target/
↳classes:/home/jhall/maven/repository/junit/junit/4.8.2/junit-4.8.2.jar:/home/jhall/
↳maven/repository/com/oracle/coherence/3.7.1.0/coherence-3.7.1.0.jar

Class path in use.....: [file:/home/jhall/maven/repository/junit/junit/4.8.2/junit-
↳4.8.2.jar]

Current system properties: {java.runtime.name=Java(TM) SE Runtime Environment, sun.
↳boot.library.path=/opt/java/jdk1.7.0_04/jre/lib/amd64, java.vm.version=23.0-b21,
↳java.vm.vendor=Oracle Corporation, java.vendor.url=http://java.oracle.com/, path.
↳separator=;, java.vm.name=Java HotSpot(TM) 64-Bit Server VM, file.encoding.pkg=sun.
↳io, idea.launcher.port=7532, user.country=GB, sun.java.launcher=SUN_STANDARD, sun.
↳os.patch.level=unknown, java.vm.specification.name=Java Virtual Machine
↳Specification, user.dir=/home/jhall/work/littlegrid/littlegrid-coherence-
↳testsupport, java.runtime.version=1.7.0_04-b20, java.awt.graphicsenv=sun.awt.
↳X11GraphicsEnvironment, java.endorsed.dirs=/opt/java/jdk1.7.0_04/jre/lib/endorsed,
↳os.arch=amd64, java.io.tmpdir=/tmp, line.separator=

Nov 12, 2012 7:56:20 AM org.littlegrid.impl.DefaultClusterMemberGroup
↳startClusterMembers
, java.vm.specification.vendor=Oracle Corporation, os.name=Linux, sun.jnu.
↳encoding=UTF-8, java.library.path=/opt/java/jdk1.7.0_04/jre/lib/amd64/server:/opt/
↳java/jdk1.7.0_04/jre/lib/amd64:/opt/java/jdk1.7.0_04/jre/./lib/amd64:/opt/java/
↳jdk1.6.0_24/jre/lib/amd64/server:/opt/java/jdk1.6.0_24/jre/lib/amd64:/opt/java/jdk1.
↳6.0_24/jre/./lib/amd64:/opt/intellij/idea-IU-117.117/bin:/usr/java/packages/lib/
↳amd64:/usr/lib64:/lib64:/lib:/usr/lib, java.specification.name=Java Platform API
↳Specification, java.class.version=51.0, sun.management.compiler=HotSpot 64-Bit
↳Tiered Compilers, os.version=3.0.0-26-generic, user.home=/home/jhall, user.
↳timezone=Europe/London, java.awt.printerjob=sun.print.PSPrinterJob, file.

↳encoding=UTF-8, idea.launcher.bin.path=/opt/intellij/idea-IU-117.117/bin, java.
↳specification.version=1.7, user.name=jhall, java.class.path=/opt/intellij/idea-IU-
↳117.117/lib/idea_rt.jar:/opt/intellij/idea-IU-117.117/plugins/junit/lib/junit-rt
↳jar:/opt/java/jdk1.7.0_04/jre/lib/deploy.jar:/opt/java/jdk1.7.0_04/jre/lib/javaws.
↳jar:/opt/java/jdk1.7.0_04/jre/lib/management-agent.jar:/opt/java/jdk1.7.0_04/jre/
↳lib/charsets.jar:/opt/java/jdk1.7.0_04/jre/lib/jsse.jar:/opt/java/jdk1.7.0_04/jre/
↳lib/jce.jar:/opt/java/jdk1.7.0_04/jre/lib/jfr.jar:/opt/java/jdk1.7.0_04/jre/lib/

```

-----
System properties before.: {java.runtime.name=Java(TM) SE Runtime Environment, sun.
↳boot.library.path=/opt/java/jdk1.7.0_04/jre/lib/amd64, java.vm.version=23.0-b21,
↳java.vm.vendor=Oracle Corporation, java.vendor.url=http://java.oracle.com/, path.
↳separator=;, java.vm.name=Java HotSpot(TM) 64-Bit Server VM, file.encoding.pkg=sun.
↳io, idea.launcher.port=7532, user.country=GB, sun.java.launcher=SUN_STANDARD, sun.
↳os.patch.level=unknown, java.vm.specification.name=Java Virtual Machine
↳Specification, user.dir=/home/jhall/work/littlegrid/littlegrid-coherence-
↳testsupport, java.runtime.version=1.7.0_04-b20, java.awt.graphicsenv=sun.awt.
↳X11GraphicsEnvironment, java.endorsed.dirs=/opt/java/jdk1.7.0_04/jre/lib/endorsed,
↳os.arch=amd64, java.io.tmpdir=/tmp, line.separator=
, java.vm.specification.vendor=Oracle Corporation, os.name=Linux, sun.jnu.
↳encoding=UTF-8, java.library.path=/opt/java/jdk1.7.0_04/jre/lib/amd64/server:/opt/
↳java/jdk1.7.0_04/jre/lib/amd64:/opt/java/jdk1.7.0_04/jre/./lib/amd64:/opt/java/
↳jdk1.6.0_24/jre/lib/amd64/server:/opt/java/jdk1.6.0_24/jre/lib/amd64:/opt/java/jdk1.
↳6.0_24/jre/./lib/amd64:/opt/intellij/idea-IU-117.117/bin:/usr/java/packages/lib/
↳amd64:/usr/lib64:/lib64:/lib:/usr/lib, java.specification.name=Java Platform API
↳Specification, java.class.version=51.0, sun.management.compiler=HotSpot 64-Bit
↳Tiered Compilers, os.version=3.0.0-26-generic, user.home=/home/jhall, user.
↳timezone=Europe/London, java.awt.printerjob=sun.print.PSPrinterJob, file.
↳encoding=UTF-8, idea.launcher.bin.path=/opt/intellij/idea-IU-117.117/bin, java.
↳specification.version=1.7, user.name=jhall, java.class.path=/opt/intellij/idea-IU-
↳117.117/lib/idea_rt.jar:/opt/intellij/idea-IU-117.117/plugins/junit/lib/junit-rt.
↳jar:/opt/java/jdk1.7.0_04/jre/lib/deploy.jar:/opt/java/jdk1.7.0_04/jre/lib/javaws.
↳jar:/opt/java/jdk1.7.0_04/jre/lib/management-agent.jar:/opt/java/jdk1.7.0_04/jre/
↳lib/charsets.jar:/opt/java/jdk1.7.0_04/jre/lib/jsse.jar:/opt/java/jdk1.7.0_04/jre/
↳lib/jce.jar:/opt/java/jdk1.7.0_04/jre/lib/jfr.jar:/opt/java/jdk1.7.0_04/jre/lib/
↳plugin.jar:/opt/java/jdk1.7.0_04/jre/lib/rt.jar:/opt/java/jdk1.7.0_04/jre/lib/
↳resources.jar:/opt/java/jdk1.7.0_04/jre/lib/ext/sunjce_provider.jar:/opt/java/jdk1.
↳7.0_04/jre/lib/ext/sunpkcs11.jar:/opt/java/jdk1.7.0_04/jre/lib/ext/zipfs.jar:/opt/
↳java/jdk1.7.0_04/jre/lib/ext/dnsns.jar:/opt/java/jdk1.7.0_04/jre/lib/ext/localedata.
↳jar:/opt/java/jdk1.7.0_04/jre/lib/ext/sunec.jar:/home/jhall/work/littlegrid/
↳littlegrid-coherence-testsupport/target/test-classes:/home/jhall/work/littlegrid/
↳littlegrid-coherence-testsupport/target/classes:/home/jhall/maven/repository/junit/
↳junit/4.8.2/junit-4.8.2.jar:/home/jhall/maven/repository/com/oracle/coherence/3.7.1.
↳0/coherence-3.7.1.0.jar, java.vm.specification.version=1.7, sun.arch.data.model=64,
↳java.home=/opt/java/jdk1.7.0_04/jre, sun.java.command=com.intellij.rt.execution.
↳application.AppMain com.intellij.rt.execution.junit.JUnit4TestRunner -ideVersion5 org.
↳littlegrid.features.jar_exclusion.JarExclusionIntegrationTest,
↳startAndShutdownWithCoherenceJarBeingExcluded, java.specification.vendor=Oracle
↳Corporation, user.language=en, awt.toolkit=sun.awt.X11.XToolkit, java.vm.info=mixed
↳mode, java.version=1.7.0_04, java.ext.dirs=/opt/java/jdk1.7.0_04/jre/lib/ext:/usr/
↳java/packages/lib/ext, sun.boot.class.path=/opt/java/jdk1.7.0_04/jre/lib/resources.
↳jar:/opt/java/jdk1.7.0_04/jre/lib/rt.jar:/opt/java/jdk1.7.0_04/jre/lib/sunrsasign.
↳jar:/opt/java/jdk1.7.0_04/jre/lib/jsse.jar:/opt/java/jdk1.7.0_04/jre/lib/jce.jar:/
↳opt/java/jdk1.7.0_04/jre/lib/charsets.jar:/opt/java/jdk1.7.0_04/jre/lib/jfr.jar:/
↳opt/java/jdk1.7.0_04/jre/classes, java.vendor=Oracle Corporation, file.separator=/,
↳java.vendor.url.bug=http://bugreport.sun.com/bugreport/, sun.cpu.endian=little, sun.
↳io.unicode.encoding=UnicodeLittle, sun.desktop=gnome, sun.cpu.isalist=}
-----

```

```

System properties applied: {tangosol.coherence.wka=127.0.0.1, tangosol.coherence.
↳site=DefaultSite, tangosol.coherence.cluster=littlegridCluster, tangosol.coherence.
↳management=none, tangosol.coherence.override=littlegrid/littlegrid-fast-start-
↳coherence-override.xml, com.sun.management.jmxremote=false, tangosol.coherence.
↳log=stdout, tangosol.coherence.ttl=0, tangosol.coherence.log.level=3, littlegrid.
↳join.timeout.milliseconds=1000, tangosol.coherence.distributed.localstorage=true,
↳tangosol.coherence.role=DedicatedStorageEnabledMember, tangosol.coherence.
↳management.remote=true, tangosol.coherence.machine=DefaultMachine, tangosol.
↳coherence.cacheconfig=coherence/littlegrid-test-cache-config.xml, tangosol.
↳coherence.rack=DefaultRack, tangosol.coherence.localport=22000, tangosol.coherence.
↳extend.address=127.0.0.1, tangosol.coherence.tcmp.enabled=true, tangosol.coherence.
↳localhost=127.0.0.1, tangosol.coherence.wka.port=22000}

```

1.22. Sample exception report

```

-----
System properties applied: sorted to help identification
  com.sun.management.jmxremote = false
  littlegrid.join.timeout.milliseconds = 1000
  tangosol.coherence.cacheconfig = coherence/littlegrid-test-cache-config.xml
  tangosol.coherence.cluster = littlegridCluster
  tangosol.coherence.distributed.localstorage = true
  tangosol.coherence.extend.address = 127.0.0.1
  tangosol.coherence.localhost = 127.0.0.1
  tangosol.coherence.localport = 22000
  tangosol.coherence.log = stdout
  tangosol.coherence.log.level = 3
  tangosol.coherence.machine = DefaultMachine
  tangosol.coherence.management = none
  tangosol.coherence.management.remote = true
  tangosol.coherence.override = littlegrid/littlegrid-fast-start-coherence-override.
↪xml
  tangosol.coherence.rack = DefaultRack
  tangosol.coherence.role = DedicatedStorageEnabledMember
  tangosol.coherence.site = DefaultSite
  tangosol.coherence.tcmp.enabled = true
  tangosol.coherence.ttl = 0
  tangosol.coherence.wka = 127.0.0.1
  tangosol.coherence.wka.port = 22000
-----
Number of members.....: 2
-----
Cluster member class name: org.littlegrid.impl.DefaultClusterMember
-----
Number of thread in pool.: 4
-----
Memory.....: max memory: 1733MB, current: 116MB, free memory: 88MB
-----
Network.....: hostname: localhost.localdomain, address: 127.0.0.1
-----
System properties current: all current system properties - sorted to help_
↪identification
  awt.toolkit = sun.awt.X11.XToolkit
  file.encoding = UTF-8
  file.encoding.pkg = sun.io
  file.separator = /
  idea.launcher.bin.path = /opt/intellij/idea-IU-117.117/bin
  idea.launcher.port = 7532
  java.awt.graphicsenv = sun.awt.X11GraphicsEnvironment
  java.awt.printerjob = sun.print.PSPrinterJob
  java.class.path = /opt/intellij/idea-IU-117.117/lib/idea_rt.jar:/opt/intellij/
↪idea-IU-117.117/plugins/junit/lib/junit-rt.jar:/opt/java/jdk1.7.0_04/jre/lib/deploy.
↪jar:/opt/java/jdk1.7.0_04/jre/lib/javaws.jar:/opt/java/jdk1.7.0_04/jre/lib/
↪management-agent.jar:/opt/java/jdk1.7.0_04/jre/lib/charsets.jar:/opt/java/jdk1.7.0_
↪04/jre/lib/jsse.jar:/opt/java/jdk1.7.0_04/jre/lib/jce.jar:/opt/java/jdk1.7.0_04/jre/
↪lib/jfr.jar:/opt/java/jdk1.7.0_04/jre/lib/plugin.jar:/opt/java/jdk1.7.0_04/jre/lib/
↪rt.jar:/opt/java/jdk1.7.0_04/jre/lib/resources.jar:/opt/java/jdk1.7.0_04/jre/lib/
↪ext/sunjce_provider.jar:/opt/java/jdk1.7.0_04/jre/lib/ext/sunpkcs11.jar:/opt/java/
↪jdk1.7.0_04/jre/lib/ext/zipfs.jar:/opt/java/jdk1.7.0_04/jre/lib/ext/dnsns.jar:/opt/
↪java/jdk1.7.0_04/jre/lib/ext/localedata.jar:/opt/java/jdk1.7.0_04/jre/lib/ext/sunec.
↪jar:/home/jhall/work/littlegrid/littlegrid-coherence-testsupport/target/test-
↪classes:/home/jhall/work/littlegrid/littlegrid-coherence-testsupport/target/
↪classes:/home/jhall/maven/repository/junit/junit/4.8.2/junit-4.8.2.jar:/home/jhall/
↪maven/repository/com/oracle/coherence/3.7.1.0/coherence-3.7.1.0.jar

```

```

java.class.version = 51.0
java.endorsed.dirs = /opt/java/jdk1.7.0_04/jre/lib/endorsed
java.ext.dirs = /opt/java/jdk1.7.0_04/jre/lib/ext:/usr/java/packages/lib/ext
java.home = /opt/java/jdk1.7.0_04/jre
java.io.tmpdir = /tmp
java.library.path = /opt/java/jdk1.7.0_04/jre/lib/amd64/server:/opt/java/jdk1.7.0_
↪04/jre/lib/amd64:/opt/java/jdk1.7.0_04/jre/../../lib/amd64:/opt/java/jdk1.6.0_24/jre/
↪lib/amd64/server:/opt/java/jdk1.6.0_24/jre/lib/amd64:/opt/java/jdk1.6.0_24/jre/../../
↪lib/amd64:/opt/intellij/idea-IU-117.117/bin:/usr/java/packages/lib/amd64:/usr/
↪lib64:/lib64:/lib:/usr/lib
java.runtime.name = Java(TM) SE Runtime Environment
java.runtime.version = 1.7.0_04-b20
java.specification.name = Java Platform API Specification
java.specification.vendor = Oracle Corporation
java.specification.version = 1.7
java.vendor = Oracle Corporation
java.vendor.url = http://java.oracle.com/
java.vendor.url.bug = http://bugreport.sun.com/bugreport/
java.version = 1.7.0_04
java.vm.info = mixed mode
java.vm.name = Java HotSpot(TM) 64-Bit Server VM
java.vm.specification.name = Java Virtual Machine Specification
java.vm.specification.vendor = Oracle Corporation
java.vm.specification.version = 1.7
java.vm.vendor = Oracle Corporation
java.vm.version = 23.0-b21
line.separator =

os.arch = amd64
os.name = Linux
os.version = 3.0.0-26-generic
path.separator = :
sun.arch.data.model = 64
sun.boot.class.path = /opt/java/jdk1.7.0_04/jre/lib/resources.jar:/opt/java/jdk1.
↪7.0_04/jre/lib/rt.jar:/opt/java/jdk1.7.0_04/jre/lib/sunrsasign.jar:/opt/java/jdk1.7.
↪0_04/jre/lib/jsse.jar:/opt/java/jdk1.7.0_04/jre/lib/jce.jar:/opt/java/jdk1.7.0_04/
↪jre/lib/charsets.jar:/opt/java/jdk1.7.0_04/jre/lib/jfr.jar:/opt/java/jdk1.7.0_04/
↪jre/classes
sun.boot.library.path = /opt/java/jdk1.7.0_04/jre/lib/amd64
sun.cpu.endian = little
sun.cpu.isalist =
sun.desktop = gnome
sun.io.unicode.encoding = UnicodeLittle
sun.java.command = com.intellij.rt.execution.application.AppMain com.intellij.rt.
↪execution.junit.JUnitStarter -ideVersion5 org.littlegrid.features.jar_exclusion.
↪JarExclusionIntegrationTest,startAndShutdownWithCoherenceJarBeingExcluded
sun.java.launcher = SUN_STANDARD
sun.jnu.encoding = UTF-8
sun.management.compiler = HotSpot 64-Bit Tiered Compilers
sun.os.patch.level = unknown
user.country = GB
user.dir = /home/jhall/work/littlegrid/littlegrid-coherence-testsupport
user.home = /home/jhall
user.language = en
user.name = jhall
user.timezone = Europe/London

```

 Builder keys and values - sorted to [help](#) identification

```

CacheConfiguration = coherence/littlegrid-test-cache-config.xml
CallbackHandlerInstanceClassName = org.littlegrid.impl.DefaultCallbackHandler
ClientCacheConfiguration =
ClientOverrideConfiguration =
ClusterMemberInstanceClassName = org.littlegrid.impl.DefaultClusterMember
ClusterName = littlegridCluster
CoreJarsToExcludeFromClassPath = alt-rt.jar,charsets.jar,deploy.jar,javaws.jar,
↪jce.jar,jfr.jar,jsse.jar,management-agent.jar,plugin.jar,resources.jar,rt.jar,dnsns.
↪jar,localedata.jar,sunec.jar,sunjce_provider.jar,sunpkcs11.jar,zipfs.jar,classes.
↪jar,ui.jar
CustomConfiguredCacheConfiguration =
CustomConfiguredClusterMemberInstanceClassName = org.littlegrid.impl.
↪DefaultClusterMember
CustomConfiguredCount = 0
CustomConfiguredRoleName = CustomConfiguredMember
ExceptionReporterInstanceClassName = org.littlegrid.impl.
↪DefaultBuildExceptionReporter
ExtendClientRoleName = ExtendClient
ExtendPort = 23000
ExtendProxyCount = 0
ExtendProxyRoleName = DedicatedExtendProxyMember
FastStartJoinTimeoutMilliseconds = 1000
JarsToExcludeFromClassPath = ,, , , coherence ,,
JmxMonitorCount = 0
JmxMonitorRoleName = DedicatedJmxMonitorMember
LogDestination = stdout
LogLevel = 3
MachineName = DefaultMachine
NumberOfThreadsInStartupPool = 4
OverrideConfiguration = littlegrid/littlegrid-fast-start-coherence-override.xml
RackName = DefaultRack
SiteName = DefaultSite
StorageDisabledClientRoleName = StorageDisabledClient
StorageEnabledCount = 2
StorageEnabledExtendProxyCount = 0
StorageEnabledExtendProxyRoleName = StorageEnabledExtendProxyMember
StorageEnabledRoleName = DedicatedStorageEnabledMember
SuggestedSleepAfterStopDuration35x = 120
SuggestedSleepAfterStopDuration36x = 3
SuggestedSleepAfterStopDurationDefault = 3
Ttl = 0
WkaAddress = 127.0.0.1
WkaPort = 22000
-----
Builder key to system property name mapping - sorted to help identification
CacheConfiguration = tangosol.coherence.cacheconfig
ClientCacheConfiguration = tangosol.coherence.cacheconfig
ClientOverrideConfiguration = tangosol.coherence.override
ClusterName = tangosol.coherence.cluster
CoherenceManagement = tangosol.coherence.management
CoherenceManagementRemote = tangosol.coherence.management.remote
CustomConfiguredCacheConfiguration = tangosol.coherence.cacheconfig
CustomConfiguredRoleName = tangosol.coherence.role
DistributedLocalStorage = tangosol.coherence.distributed.localstorage
ExtendAddress = tangosol.coherence.extend.address
ExtendClientRoleName = tangosol.coherence.role
ExtendEnabled = tangosol.coherence.extend.enabled
ExtendPort = tangosol.coherence.extend.port

```



```

ExtendProxyRoleName = tangosol.coherence.role
FastStartJoinTimeoutMilliseconds = littlegrid.join.timeout.milliseconds
JmxMonitorRoleName = tangosol.coherence.role
LocalAddress = tangosol.coherence.localhost
LocalPort = tangosol.coherence.localport
LogDestination = tangosol.coherence.log
LogLevel = tangosol.coherence.log.level
MachineName = tangosol.coherence.machine
ManagementJmxRemote = com.sun.management.jmxremote
OverrideConfiguration = tangosol.coherence.override
RackName = tangosol.coherence.rack
SiteName = tangosol.coherence.site
StorageDisabledClientRoleName = tangosol.coherence.role
StorageEnabledExtendProxyRoleName = tangosol.coherence.role
StorageEnabledRoleName = tangosol.coherence.role
TcmpEnabled = tangosol.coherence.tcmp.enabled
Ttl = tangosol.coherence.ttl
WkaAddress = tangosol.coherence.wka
WkaPort = tangosol.coherence.wka.port

```

Full exception.....: detailed below

```

CHECK_CHILD_FIRST_CLASS_PATH_IN_USE There were 2 member ids [1, 1] - however only_
↳these were unique member ids [1]. Ensure that the Coherence JAR is on your test_
↳class path
  at org.littlegrid.impl.DefaultClusterMemberGroup.
↳ensureMemberIdsAreUnique (DefaultClusterMemberGroup.java:257)
  at org.littlegrid.impl.DefaultClusterMemberGroup.
↳startClusterMembers (DefaultClusterMemberGroup.java:232)
  at org.littlegrid.impl.DefaultClusterMemberGroupBuilder.
↳buildStorageEnabledMembers (DefaultClusterMemberGroupBuilder.java:574)
  at org.littlegrid.impl.DefaultClusterMemberGroupBuilder.
↳buildClusterMembers (DefaultClusterMemberGroupBuilder.java:399)
  at org.littlegrid.impl.DefaultClusterMemberGroupBuilder.
↳build (DefaultClusterMemberGroupBuilder.java:297)
  at org.littlegrid.impl.DefaultClusterMemberGroupBuilder.
↳buildAndConfigureForStorageDisabledClient (DefaultClusterMemberGroupBuilder.java:320)
  at org.littlegrid.features.jar_exclusion.JarExclusionIntegrationTest.
↳startAndShutdownWithCoherenceJarBeingExcluded (JarExclusionIntegrationTest.java:66)
  at sun.reflect.NativeMethodAccessorImpl.invoke0 (Native Method)
  at sun.reflect.NativeMethodAccessorImpl.invoke (NativeMethodAccessorImpl.java:57)
  at sun.reflect.DelegatingMethodAccessorImpl.invoke (DelegatingMethodAccessorImpl.
↳java:43)
  at java.lang.reflect.Method.invoke (Method.java:601)
  at org.junit.runners.model.FrameworkMethod$1.runReflectiveCall (FrameworkMethod.
↳java:44)
  at org.junit.internal.runners.model.ReflectiveCallable.run (ReflectiveCallable.
↳java:15)
  at org.junit.runners.model.FrameworkMethod.invokeExplosively (FrameworkMethod.
↳java:41)
  at org.junit.internal.runners.statements.InvokeMethod.evaluate (InvokeMethod.
↳java:20)
  at org.junit.internal.runners.statements.ExpectException.evaluate (ExpectException.
↳java:21)
  at org.junit.internal.runners.statements.RunAfters.evaluate (RunAfters.java:31)
  at org.junit.runners.BlockJUnit4ClassRunner.runNotIgnored (BlockJUnit4ClassRunner.
↳java:79)
  at org.junit.runners.BlockJUnit4ClassRunner.runChild (BlockJUnit4ClassRunner.
↳java:71)

```

```
at org.junit.runners.BlockJUnit4ClassRunner.runChild(BlockJUnit4ClassRunner.  
↪ java:49)  
at org.junit.runners.ParentRunner$3.run(ParentRunner.java:193)  
at org.junit.runners.ParentRunner$1.schedule(ParentRunner.java:52)  
at org.junit.runners.ParentRunner.runChildren(ParentRunner.java:191)  
at org.junit.runners.ParentRunner.access$000(ParentRunner.java:42)  
at org.junit.runners.ParentRunner$2.evaluate(ParentRunner.java:184)  
at org.junit.runners.ParentRunner.run(ParentRunner.java:236)  
at org.junit.runner.JUnitCore.run(JUnitCore.java:157)  
at com.intellij.junit4.JUnit4IdeaTestRunner.  
↪ startRunnerWithArgs(JUnit4IdeaTestRunner.java:76)  
at com.intellij.rt.execution.junit.JUnitStarter.  
↪ prepareStreamsAndStart(JUnitStarter.java:195)  
at com.intellij.rt.execution.junit.JUnitStarter.main(JUnitStarter.java:63)  
at sun.reflect.NativeMethodAccessorImpl.invoke0(Native Method)  
at sun.reflect.NativeMethodAccessorImpl.invoke(NativeMethodAccessorImpl.java:57)  
at sun.reflect.DelegatingMethodAccessorImpl.invoke(DelegatingMethodAccessorImpl.  
↪ java:43)  
at java.lang.reflect.Method.invoke(Method.java:601)  
at com.intellij.rt.execution.application.AppMain.main(AppMain.java:120)  
-----  
Possible exception reason: CHECK_CHILD_FIRST_CLASS_PATH_IN_USE There were 2 member_  
↪ ids [1, 1] - however only these were unique member ids [1]. Ensure that the_  
↪ Coherence JAR is on your test class path
```

littlegrid JavaDoc

The JavaDoc is presently unavailable whilst hosting is being sorted out - apologies.

littlegrid test coverage report

Note: littlegrid has an extensive set of unit and integration/functional, test coverage of littlegrid is currently over 98%.

The test coverage reports are presently unavailable whilst hosting is being sorted out - apologies.

FAQ

Got a question or having a problem that isn't answered below, then please feel free to contact me: [help at littlegrid dot org](mailto:help@littlegrid.org)

- 17. How does littlegrid work?
 - 1. littlegrid loads each Coherence cluster member into its own 'child-first' class-loader using the settings that littlegrid has defined for cluster ports and names, this ensures that you don't have any accidental clustering.

Note: littlegrid does rely on using Coherence system properties to control certain aspects of Coherence's behaviour, such as the Extend proxy address and port, along with others.

- 17. How many cluster members can I run in a single JVM?
 - 1. Quite a few! On an average laptop/desktop, it is perfectly easy to run 30 or more cluster members in a single JVM by setting the memory settings appropriately.
- 17. I can't seem to start my cluster with littlegrid.
 - 1. In the event a problem starting up your cluster, then littlegrid will produce an 'exception report' *Sample exception report* which details the class path, system properties and other useful information to help diagnose the problem. If you're really stuck, then just email me the exception report and I'll take a look :-)
- 17. I'm getting perm-space problems.
 - 1. This is because each Coherence cluster member gets loaded into their own class-loader, this causes a lot of classes to be loaded, with the Sun JVM, these classes reside in the perm-space. To accommodate the extra classes, simply include the following as it will be fine for most cases:


```
-XX:MaxPermSize=256m
```
- 17. I'm using Ant and it looks like it doesn't load some littlegrid classes - I'm getting ClassNotFoundExceptions.
 - 1. When using Ant, please set JUnit to fork:


```
<junit fork="yes">
  ...
</junit>
```
- 17. I'm using Maven and getting perm-space problems.
 - 1. With Maven, you need to configure the Surefire plugin, take a look at the *Maven best practice*
- 17. I'm seeing java.lang.SecurityException: Prohibited package name: java.lang- why is that?
 - 1. Please upgrade to version 2.12 (or later) - older versions of littlegrid used your JAVA_HOME environment variable to work out which core Java JAR files (such as rt.jar) should be excluded from its child-first class loader - with 2.12 the requirement for JAVA_HOME to match your IDE is no longer required.
- 17. I'm seeing an exception message with Error instantiating Filter with name: gzip - why is that?
 - 1. When used with littlegrid, Coherence 3.7.0.x onwards gives this exception if none of your cache services are marked as autostart. As of version 2.13, littlegrid attempts to identify this exception and report it as requiring a service to be auto-started.


```
<autostart>true</autostart>
```

Support

- Please refer to the [FAQ](#).
- When an exception occurs during littlegrid start-up it produces an exception report (*Sample exception report*), this is useful for troubleshooting.
- littlegrid maintains an active [issue tracking](#) on BitBucket

Note: if you're stuck, then please feel free to email the generated exception report to help at [littlegrid dot org](mailto:littlegrid@dot.org)

License

littlegrid is an open source project and has a BSD license (see below)

```
Copyright (c) 2010-2013 Jonathan Hall.  
All rights reserved.
```

```
Redistribution and use in source and binary forms, with or without  
modification, are permitted provided that the following conditions are met:
```

- Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
- Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

```
Neither the name of the littlegrid nor the names of its contributors may  
be used to endorse or promote products derived from this software without  
specific prior written permission.
```

```
THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS"  
AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE  
IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE  
ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE  
LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR  
CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF  
SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS  
INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER  
IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE)  
ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE  
POSSIBILITY OF SUCH DAMAGE.
```

Release notes

2013-07-14 Version 2.15.2

- #135 Add workaround for 12.1.2.0.0 DefaultCacheServer.shutdown NPE when static start is used

2013-05-14 Version 2.15.1

- #126: Simple keep-alive cluster member group should always report current usage count higher than 1 to ensure it doesn't get shutdown

2013-05-09 Version 2.15

- #85: Enable integration tests requiring the same type of cluster member group to share the started group, such as via a holder
- #105 Add timeout milliseconds to identifiable exception
- #106: Provide an interactive console for stopping, shutting down and merging cluster members
- #108: Change Coherence GAV co-ordinates to be in-line with 12c
- #109: Create console package and move simple wait console into package
- #114: Add the ability to have a specific cache configuration file for storage enabled members

- #115: Add the ability to have a specific cache configuration file for Extend proxy members
- #116: Add the ability to have a specific cache configuration file for JMX monitor members
- #117: Add the ability to have a specific log level for Extend proxy, storage enabled members and clients (Extend and storage-disabled)
- #118: Combination of Windows and System.getenv in certain environments returns keys that are in uppercase - even through the environment variable is mixed case
- #119: Include ability to sleep until a particular time in the command DSL
- #121: Make logging more compact, i.e. fewer lines
- #122: Tidy up fast-start and builder interaction
- #125: Add ability to specify site, rack and machine within the command DSL shell

2013-01-28 Version 2.14

- #100: Expand the concept of a command-line launcher to accept a string which equates to a build enum, for clients launch a console
- #104: Add build method accepting an enum in order allow the cluster member group to be built
- #103: Add WKA and Extend port getters on cluster member group to enable API to be more fluent when dealing with multiple clusters
- #102: Add reminder(s) for people to check the littlegrid web-site for updated information and new versions

2012-11-08 Version 2.13

- #99: Add a simple launcher application to enable a littlegrid cluster to be started from the command-line or via .Net or similar
- #91: Add functionality to more easily control specified cluster members via the actual class loader
- #93: Change system environment variable prefix to littlegrid_builder_X and system property prefix to littlegrid.builder.X instead of littlegrid.builder.override.X. Keep original prefix working
- #46: Add support for client override to be set via builder and then applied before client starts
- #92: Add additional convenience methods for setting additional system properties, for instance accepting an integer and boolean as the value
- #98: When no caches have their service marked as auto-start and Coherence 3.7.0.x or later are used then exception occurs: Error instantiating Filter with name: gzip

2012-09-23 Version 2.12

- #90: Add classes.jar and ui.jar for core JAR exclusion

2012-09-20 Version 2.11

- #65: Add functionality to report possible reasons for a security exception when performing child-first class loading
- #83: Improve the way that littlegrid excludes core Java JAR files from its class-path used by the child-first class loader
- #86: Ensure that resources are loaded into child-first class loader if possible to ensure they are isolated
- #88: When an exception occurs loading a class with the child first class loader, output the name of the class that it was trying to load
- #89: Exclude any classes with "java." from child-first class loading

2012-08-21 Version 2.10

- #80: Child first URL class loader's loadClass is susceptible to a race condition in certain use cases

2012-08-08 Version 2.9

- #81: If a security exception occurs then throw a meaningful exception to help user to identify and fix the problem
- #80: Child first URL class loader's loadClass is susceptible to a race condition in certain use cases

2012-05-04 Version 2.8

- #73: Add warning message when trying to perform a stop or shutdown on a non-existent member
- #74: Add support for setting a system property easily via the builder
- #75: Improve JMX test example
- #76: Add support for site, rack and machine within builder

2012-04-13 Version 2.7

- #7: Include simple contains check for JAR exclusion.
- #52: Bake in the littlegrid version number and display on start-up.
- #66: Expose the functionality to allow merging of cluster member groups.
- #67: For Maven builds use surefire.test.class.path instead of java.class.path.
- #68: Add support for overriding properties using environment variables and system properties.
- #70: Add getter for WKA address.
- #71: Change default WKA and Extend port numbers to increase 'gap' between them.

2012-03-26 Version 2.6.1

- #51: In the event of an exception on start-up, output the builder keys and values in the exception report.
- #55: Remove logger place holder - just use Java util logging for now.
- #56: Reduce log level when reading default properties files.
- #58: Make the Extend-related invocables used in test pack public for other people to use.
- #59: Change setBuilderProperties to accept multiple properties files and multiple names of properties file.
- #60: Remove explicit WKA port numbers from builder setting test.
- #62: Add concept of post cluster member group start support.
- #63: Add builder for storage-enabled member.
- #64: Change invocation of doAfterStart of cluster member group, so that it occurs after the 'client' environment has been configured.

2012-02-28 Version 2.5.1

- #4: Enable mapping of Builder bean setter names to different system properties.
- #44: Implement basic (sequential) shutdown for multiple members.
- #45: Implement basic (sequential) stop for multiple members.
- #47: Change default behaviour so that fast-start is disabled by default.
- #48: Add support for custom-configured to specify their own cache configuration within a cluster member group.
- #49: Remove the defaulting to 1 storage-enabled member if no specific member types are set - i.e. make it more explicit and require to be set.

- #50: Add specific build methods so that it is clearer to a user what the ‘environment’ is being set as ready for their tests. Examples being storage-disabled client or Extend client.

2012-02-19 Version 2.4.8

- #40: Make project available via Nexus OSS

2011-02-06 Version 2.2.3

- #1: Enable overriding builder properties using environment variable and properties file.
- #3: Add mapping from Builder bean setter names to Coherence system properties.
- #5: Add JMX support.
- #6: Tidy up builder.
- #8: Look at why Coherence 3.5 reflection updater doesn’t work with integers (and so need to use strings).
- #9: Add checks for incoming parameters for DefaultLocalProcessClusterMemberGroup constructor.
- #11: Add support for more than one than one Extend proxy in a cluster member group containing dedicated Extend proxies.
- #12: Add support for more than one than one Extend proxy in a cluster member group containing storage enabled Extend proxies.
- #13: Fix startAndShutdownWithKnownRequiredJarBeingExcluded test so that it doesn’t use multicast (it does so because member group is shutdown and resets the system properties).
- #15: Improve builder setBuilderProperties to take a string for a properties filename, this will avoid the API user having to worry about getResourceAsStream etc.
- #16: Add getExtendPort method to builder to allow this to be used as the starting port for people to offset from it.
- #18: Move code over to common resource loading for properties be it from the filesystem or class path.
- #19: Add license to all source code.
- #21: Change system property for LITTLEGRID_COHERENCE_OVERRIDE to use dots instead of hyphens.
- #23: Add tests for explicit setting of override file.
- #24: Reduce the size of the ‘public’ API - ensure minimum and primarily interfaces.
- #26: On builder, provide method to add system properties.
- #27: Add test for Extend proxy stop.
- #28: DefaultClusterMemberGroupBuilder.setClusterName() unimplemented.
- #29: Reflection based cluster member with fallback to default behaviour.
- #32: Before starting to build the cluster group output exactly what the final group will contain.
- #33: Add ‘fast-start’ override file that reduces delay time before creating cluster.
- #34: Add build incident reporter.
- #35: Add support for the properties utils to log at a particular level.
- #36: Add builder style support to reflection delegating cluster member.
- #37: Improve JavaDoc for usage of stop and sleep.
- #38: JMX monitor member doesn’t start-up when an Extend proxy is also used - system property not cleared causing problem.

- #41: Reset system properties after cluster is up.

2011-11-13 Version 2.0.2

2011-11-02 Version 1.13

2011-08-03 Version 1.12

2011-07-27 Version 1.11

2011-06-28 Version 1.0.6

2011-05-30 Version 1.0.5

2011-05-25 Version 1.0.4

2011-04-06 Version 1.0.3

2011-02-22 Version 1.0.1

2011-01-24 Version 1.0

2010-10 Version 0.1 * Development of tumbleweed started - this was the original name for littlegrid.

Management REPL application

The management REPL application is a new concept that is under development in littlegrid 2.16-SNAPSHOT and is therefore subject to change or possible removal.

If you do want to play with the management REPL app, then it can be started using a command similar to below (changing the UrlPath to match the connection string to your JMX monitor node):

```
java -Dlittlegrid.management.builder.UrlPath="service:jmx:rmi:///jndi/rmi://
↳localhost:50003/jmxrmi" -cp coherence.jar;littlegrid-2.16-SNAPSHOT.jar org.
↳littlegrid.app.ManagementReplApp
```

Once the console has started you can then invoke the following commands (it is envisaged that the commands available will be expanded in the future based upon feedback):

- select columns from MBean where column = someValue

```
select min(get('PacketsSent')), max(get('PacketsSent')), sum(get('PacketsSent')) from
↳'Coherence:type=Node,nodeId=*
```

Resulting in (output to be tweaked/improved):

```
{DoubleMin(.get(PacketsSent))=1922.0, DoubleMax(.get(PacketsSent))=99404.0,
↳DoubleSum(.get(PacketsSent))=113064.0}
```

```
select get('nodeId'), get('service'), get('name'), get('Size') from
↳'Coherence:type=Cache,service=*,name=*,nodeId=*,tier=back' where get('StoreFailures
↳') > 0L
```

```
select get('Units'), count() from 'Coherence:type=Cache,service=DistributedCache,
↳name=test,nodeId=*,tier=back' group by get('Units')
```

- create snapshot MySnapshot MBean

```
create snapshot node 'Coherence:type=Node,nodeId=*
```


A snapshot is used instead of specifying the MBean:

```
select get('nodeId'), get('PacketsSent') from node
```

- show snapshots

```
show snapshots
```

- drop snapshot MySnapshot

```
drop snapshot node
```

Symbols

.Net

- Extend proxy, 24
- REPL and Batch command apps, 22

A

API

- Controlling the builder, 42

B

BSD

- littlegrid license, 55

Builder

- Mapping to system properties, 44
- Value mapping, 42

C

C++

- Extend proxy, 24
- REPL and Batch command apps, 22

Continuous integration, 40

D

Documentation

- Updates - recent changes to pages, 3

Download

- Getting littlegrid JAR, 6

DSL

- Batch command app, 22
- REPL Apps, 22

E

Entry processor

- Entry processor failover example, 25

Environment variables

- Controlling the builder, 42

Exception report

- Sample output, 47

Extend

- JUnit example, 10

F

Failover

- Entry processor example, 25

Fast-start join timeout

- Faster, 20

Faster

- Fast-start option, 20
- Reusable cluster member group, 20

G

Getting started

- Simple example, 7

I

Improve test build time

- Reusable cluster member group, 46

J

JUnit

- Extend example, 10
- Simple example, 7

L

License

- BSD, 55

littlegrid-builder-default.properties

- Controlling the builder, 42

littlegrid-builder-override.properties

- Controlling the builder, 42

M

Maven

- Best practice, 39
- Dependency, 39
- Surefire plugin, 39

Mini-cluster

- example, 24

Multiple separate clusters, 38

P

Parallel tests

 Faster, 46

Perm-space problems, 54

R

Release notes, 56

Reusable cluster member group

 Improve build test time, 46

S

Shutdown cluster member

 Entry processor example, 25

SimpleKeepAliveClusterMemberGroup

 Improve build test time, 46

Source code

 Getting the littlegrid source code, 6

Stop cluster member

 Entry processor example, 25

Support

 Sending the exception report, 47

 Trouble shooting and getting help, 55

Surefire

 Memory configuration, 39

System properties

 Controlling the builder, 42

 Mapping to builder, 44

T

Test coverage, 54

Trouble shooting

 Using the exception report, 47

U

UsageCountingClusterMemberGroup

 Improve build test time, 46

Users

 Who is using littlegrid, 5

W

Who is using

 littlegrid users, 5

WKA port

 Example using multiple clusters, 38